

# Báo cáo

## Thành viên

Nhóm gồm 2 sinh viên:

1. Lê Ngọc Hải - MSSV: 15520182
2. Lê Văn Hạnh - MSSV: 15520197

## Thu thập Dữ liệu

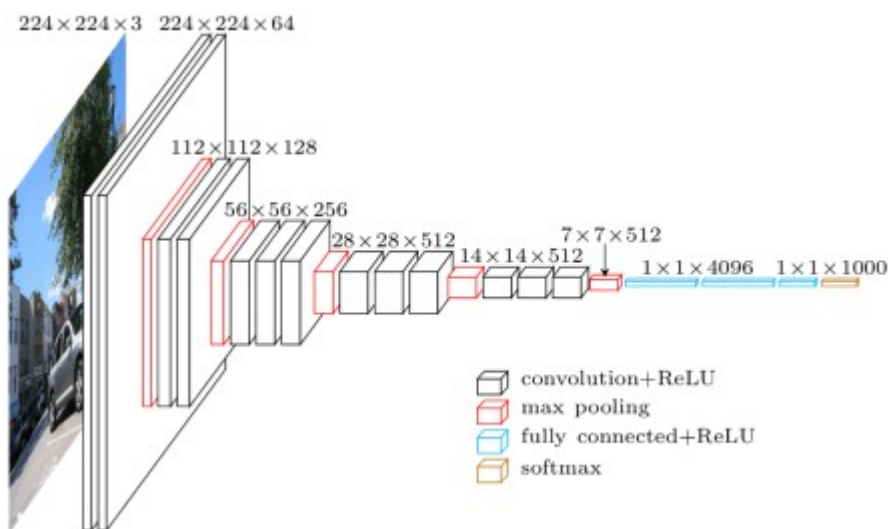
Nhóm thu thập được 30 tấm ảnh, chia đều cho hai lớp đối tượng như đã yêu cầu. Đối tượng mà nhóm chọn là: **motobike** và **non-motobike**.

**Thông tin dữ liệu:** Dữ liệu này gồm những ảnh được chụp thực tế ở trong khuôn viên Kí túc xá ĐHQG khu B.

Tiếp theo, là bước rút trích đặc trưng của dữ liệu.

## Rút trích dữ liệu

Đặc trưng của ảnh được rút trích sử dụng mạng CNN VGG16 được đề cập ở paper Very Deep Convolutional Networks for Large-Scale Image Recognition. Có 2 phiên bản phổ biến là VGG16 với 16 layer và VGG19 với 19 layer. Kiến trúc của mạng VGG16 tuy nhiều lớp nhưng khá đơn giản bao gồm 5 block convolutional với 13 lớp và 3 lớp fully-connected layer.



Chương trình sử dụng ma trận kết quả của lớp FC thứ nhất (nằm ở block thứ 6 nên được gọi là FC6) làm đặc trưng cho ảnh. Các tham số của mạng được lấy từ file của tác giả paper train trên tập ImageNet với 1000 lớp. Mạng có thể được train trên tập dữ liệu 30 ảnh tự thu thập nhưng số lượng ảnh quá ít nên thích hợp cho một mô hình deep-learning.

Có nhiều phiên bản implement của các tác giả khác nhau. Qua thử nghiệm, phiên bản cài đặt mạng VGG16 của thư viện keras-application phù hợp với máy tính cá nhân nên được sử dụng ở đây. Ảnh đầu vào phải có kích thước 224x224x3 nếu sử dụng tham số được train từ tập ImageNet. Kết quả đầu ra của model là ma trận kết quả của lớp FC thứ nhất được làm phẳng thành một vector và sử dụng làm input cho mô hình SVM.

## Xây dựng mô hình

Nhóm sử dụng mô hình Linear SVM của thư viện scikit-learn để phân loại hai lớp đối tượng trên tập train.

Khai báo thư viện:

```
from sklearn import svm
```

Khai báo mô hình:

```
clf = svm.LinearSVC()
```

Tham số huấn luyện là: `LinearSVC(penalty='l2', loss='squared_hinge', dual=True, tol=1e-4, C=1.0, multi_class='ovr', fit_intercept=True, intercept_scaling=1, class_weight=None, verbose=0, random_state=None, max_iter=1000)`

Học trên dữ liệu train:

```
clf.fit(X_train_fc6, y_train)
```

Lưu lại mô hình nhóm dùng thư viện pickle của Python.

Khai báo thư viện:

```
import pickle
```

Đặt tên file sẽ lưu là **model.sav**:

```
filename = 'model.sav'
```

Lưu file mô hình đã học xuống đĩa:

```
pickle.dump(clf, open(filename, 'wb'))
```

## Test dữ liệu trên mô hình đã lưu

Load file mô hình đã lưu:

```
loaded_model = pickle.load(open(filename, 'rb'))
```

Test trên dữ liệu và ghi kết quả ra màn hình:

```
scores = loaded_model.score(X_test_fc6, y_test)
```

```
print("accuracy = %f" %scores)
```

Kết quả test trên dữ liệu là: **100%**

## **Hướng dẫn chạy chương trình:**

Chỉ cần chạy file python main.py là được.

*Command line:* **python3 main.py**

Link github: <https://github.com/lvhanh270597/AdvancedML>

## **Tài liệu tham khảo**

[https://drive.google.com/file/d/17oW3XdSWpK\\_23KN\\_Un4eEN1BuEZLnuq\\_/view?usp=sharing](https://drive.google.com/file/d/17oW3XdSWpK_23KN_Un4eEN1BuEZLnuq_/view?usp=sharing)

<https://machinelearningmastery.com/save-load-machine-learning-models-python-scikit-learn/>

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html#sklearn.svm.LinearSVC>