

XIN CHÀO!

TÔI LÀ TRƯƠNG VĂN NHẤT

MSSV: 16521759

Email: vannhat820@gmail.com





NHẬN DIỆN BIỂU BẢO GIAO THỐNG VỚI CNN

GVHD: PGS.TS LÊ ĐÌNH DUY

MÔN HỌC: CS519.L11

NỘI DUNG TRÌNH BÀY

- I. Giới thiệu chung
- II. Chuẩn bị dữ liệu
- III. Thuật toán CNN
- IV. Hướng tiếp cận
- V. Thực nghiệm
- VI. Tài liệu tham khảo

I. Giới thiệu chung

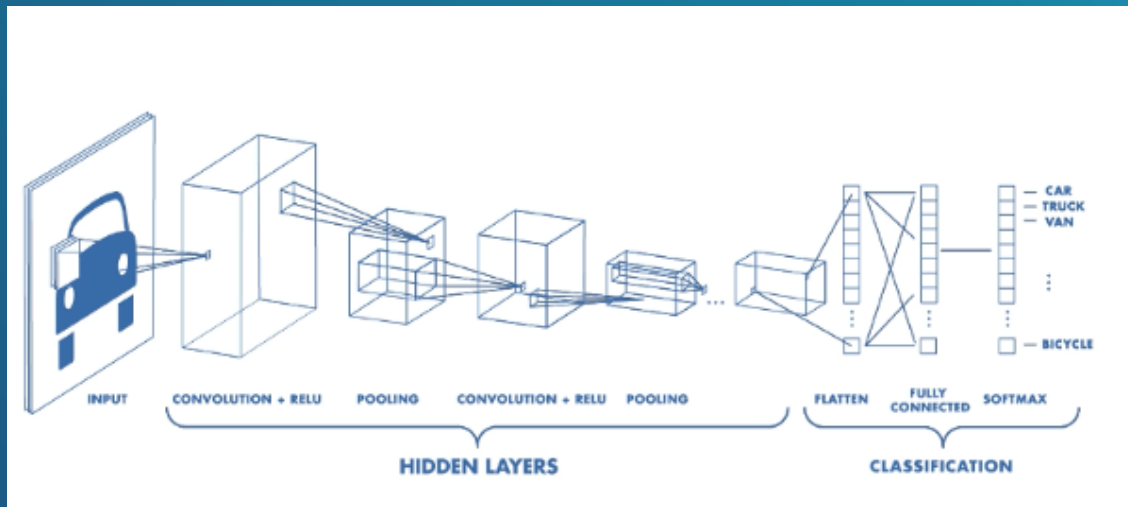
- ◆ Tìm hiểu và cài đặt thuật toán CNN để ứng dụng vào nhận diện biển báo giao thông ở Việt Nam.
- ◆ Input: Ảnh có chứa biển báo giao thông và cảnh vật.
- ◆ Output: Xác định được vùng biển báo giao thông và đưa ra thông tin về biển báo giao thông đó.
- ◆ Lý do chọn đề tài: Ứng dụng được công nghệ thông tin để giải quyết các vấn đề trong lĩnh vực giao thông.

II. Chuẩn bị dữ liệu

- ◆ Nhóm sử dụng bộ dữ liệu GTSRB - German Traffic Sign Recognition Benchmark gồm 43 loại biển báo của Đức.
- ◆ Trong bài báo cáo này nhóm chỉ giới hạn nhận dạng 16 loại biển báo nhưng được gộp thành 4 nhóm chính đó là: Biển báo đường cấm, biển báo cấm đi ngược chiều, biển báo dừng, nhóm biển báo tốc độ tối đa cho phép.
- ◆ Như vậy tập dữ liệu của chúng ta bao gồm khoảng 20.000 bức ảnh của 16 loại biển báo. Mỗi loại biển báo có khoảng hơn 1.000 tấm với đầy đủ kích thước khác nhau. Và tập dataset này sẽ được chia thành hai phần đó là training và validation theo tỷ lệ training/validation là 0.8/0.2

III. Thuật toán CNN

Cấu trúc mạng Convolutional Neutral Network



III. Thuật toán CNN

- ◆ Convolutional Neural Networks (CNN) là một trong những mô hình deep learning phổ biến nhất và có ảnh hưởng nhiều nhất trong cộng đồng Computer Vision. CNN được dùng trong nhiều bài toán như nhận dạng ảnh, phân tích video, ảnh MRI, hoặc cho bài các bài của lĩnh vực xử lý ngôn ngữ tự nhiên, và hầu hết đều giải quyết tốt các bài toán này.
- ◆ Mô hình CNN có 2 khía cạnh cần quan tâm là tính bất biến (Location Invariance) và tính kết hợp (Compositionality). Với cùng một đối tượng, nếu đối tượng này được chiếu theo các góc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể.

III. Thuật toán CNN

So với các thuật toán thông thường, CNN hiệu quả hơn hẳn bởi vì:

- ◆ Weight sharing: Trong các convolutional layers, khi thực hiện nhân tích chập sẽ dùng cùng các trọng số như nhau. Một kernel có thể dùng được nhiều lần, trong một bức ảnh.
- ◆ Các convolutional layer giúp trích xuất đặc trưng của một tấm ảnh/video, nhờ vậy mạng nơ-ron tích chập có khả năng học được các đặc trưng này (Feature learning).
- ◆ Pre-trained model: Có thể dùng lại được một model đã được train trước đó cho một bài toán hoàn toàn mới.

III. Thuật toán CNN

Cách chọn tham số cho CNN

- ◆ Số các convolution layer: càng nhiều các convolution layer thì performance càng được cải thiện. Sau khoảng 3 hoặc 4 layer, các tác động được giảm một cách đáng kể.
- ◆ Filter size: thường filter theo size 5×5 hoặc 3×3 .
- ◆ Pooling size: thường là 2×2 hoặc 4×4 cho ảnh đầu vào lớn.
- ◆ Cách cuối cùng là thực hiện nhiều lần việc train test để chọn ra được param tốt nhất.

IV. Hướng tiếp cận

1. Training model

```
#Splitting the images into train and validation sets
(X_train,X_val)=Cells[(int)(0.2*len(labels)):],Cells[::(int)(0.2*len(labels))]
X_train = X_train.astype('float32')/255
X_val = X_val.astype('float32')/255
(y_train,y_val)=labels[(int)(0.2*len(labels)):],labels[::(int)(0.2*len(labels))]

#Using one hote encoding for the train and validation labels
from keras.utils import to_categorical
y_train = to_categorical(y_train, 43)
y_val = to_categorical(y_val, 43)
```

Tiền xử lý và phân chia dữ liệu

IV. Hướng tiếp cận

1. Training model

```
#Definition of the DNN model

from keras.models import Sequential
from keras.layers import Conv2D, MaxPool2D, Dense, Flatten, Dropout

model = Sequential()
model.add(Conv2D(filters=32, kernel_size=(5,5), activation='relu', input_shape=X_train.shape[1:]))
model.add(Conv2D(filters=32, kernel_size=(5,5), activation='relu'))
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(rate=0.25))
model.add(Conv2D(filters=64, kernel_size=(3, 3), activation='relu'))
model.add(Conv2D(filters=64, kernel_size=(3, 3), activation='relu'))
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(rate=0.25))
model.add(Flatten())
model.add(Dense(256, activation='relu'))
model.add(Dropout(rate=0.5))
model.add(Dense(43, activation='softmax'))
```

Định nghĩa mô hình CNN

IV. Hướng tiếp cận

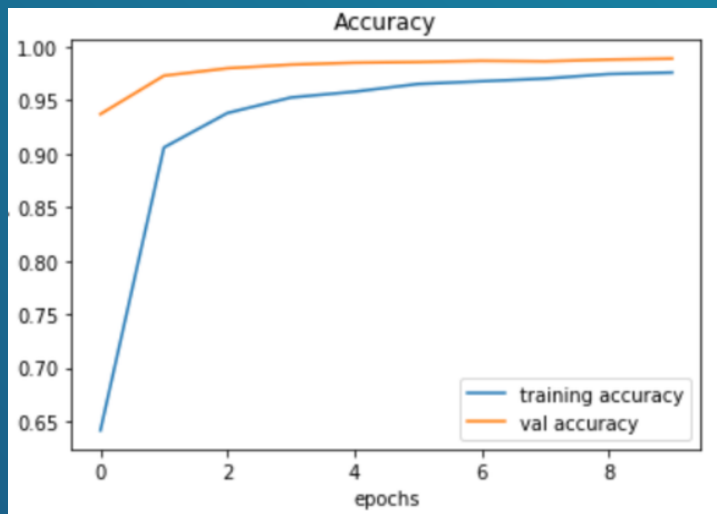
1. Training model

```
Train on 31368 samples, validate on 7841 samples
Epoch 1/10
31368/31368 [=====] - 9s 286us/step - loss: 1.2740 - acc:
Epoch 2/10
31368/31368 [=====] - 5s 175us/step - loss: 0.3047 - acc:
Epoch 3/10
31368/31368 [=====] - 6s 204us/step - loss: 0.1999 - acc:
Epoch 4/10
31368/31368 [=====] - 6s 199us/step - loss: 0.1576 - acc:
Epoch 5/10
31368/31368 [=====] - 5s 171us/step - loss: 0.1335 - acc:
Epoch 6/10
31368/31368 [=====] - 5s 171us/step - loss: 0.1175 - acc:
Epoch 7/10
31368/31368 [=====] - 5s 171us/step - loss: 0.1043 - acc:
Epoch 8/10
31368/31368 [=====] - 5s 172us/step - loss: 0.0962 - acc:
Epoch 9/10
31368/31368 [=====] - 5s 172us/step - loss: 0.0849 - acc:
Epoch 10/10
31368/31368 [=====] - 5s 172us/step - loss: 0.0820 - acc:
```

Tiến hành training model

IV. Hướng tiếp cận

1. Training model



Kiểm tra độ chính xác của model

IV. Hướng tiếp cận

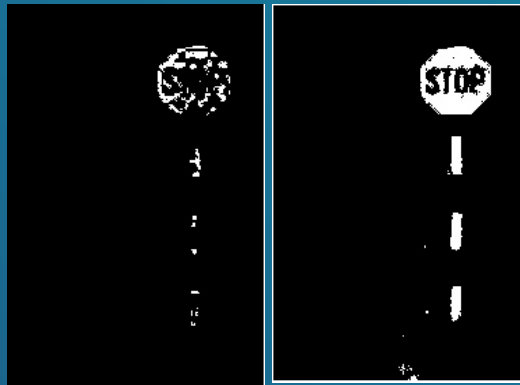
2. Phân đoạn ảnh



Ảnh đầu vào, Mark 1

IV. Hướng tiếp cận

2. Phân đoạn ảnh



Mark 2, Phân vùng biển báo

IV. Hướng tiếp cận

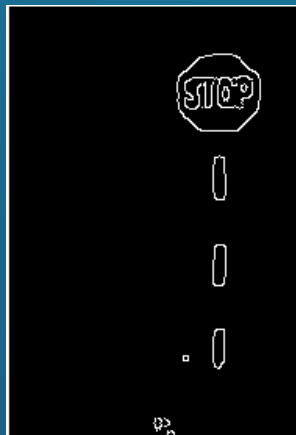
2. Phân đoạn ảnh



Mark được làm mờ và mịn

IV. Hướng tiếp cận

3. Phát hiện vùng ứng viên và lọc nhiễu



Kết quả tìm biên của đối tượng, Khoanh vùng ứng viên

IV. Hướng tiếp cận

3. Phát hiện vùng ứng viên và lọc nhiễu



Khoang vùng biển báo, Kết quả crop biển báo từ ảnh đầu vào



THỰC NGHIỆM

Search Environments



Installed

Channels

Update index...

Search Packages



C:\Windows\system32\cmd.exe - python manage.py runserver

(opencv) C:\Users\NhatTV>E:

(opencv) E:\>cd CS114.L11-Traffic-Sign-Recognition\18521175-18520424-16521759

(opencv) E:\CS114.L11-Traffic-Sign-Recognition\18521175-18520424-16521759>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

January 04, 2021 - 21:49:25

Django version 3.1.4, using settings 'trafficsign.settings'

Starting development server at http://127.0.0.1:8000/

Quit the server with CTRL-BREAK.

Version

2.3.0

0.11.0

3.7.3

3.3.1

1.6.3

3.0.1

20.3.0

1.0

1.4

1.20.1

1.0.9

0.7.0

1.0.8



Create



Clone



Import



Remove



bzip2



High-quality data compressor

145 packages available

VI. TÀI LIỆU THAM KHẢO

1. Diễn đàn Cấp cao Công nghệ Thông tin – Truyền thông Việt Nam (Vietnam ICT Summit), 2015. CNTT và Quản trị thông minh.
2. <https://www.kaggle.com/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign>
3. <https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/>

THANKS!

ANY QUESTIONS?

