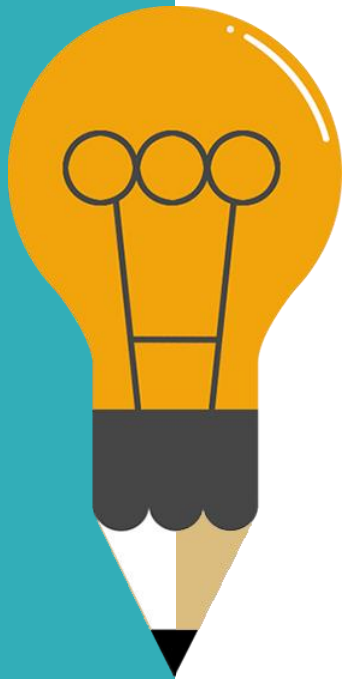


Nhập môn Thị giác máy tính

Đề Tài : Lái xe tự động trong game
bằng Deeplearning

Giảng viên hướng dẫn: Thầy Mai Tiến Dũng
Sinh Viên : Nguyễn Vũ Dương - 20520465

BỒ CỤC



01

GIỚI THIỆU

02

CƠ SỞ LÝ THUYẾT

03

THIẾT KẾ HỆ THỐNG

04

KẾT QUẢ

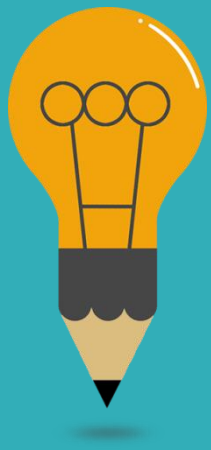
05

KẾT LUẬN



CHƯƠNG 1:

GIỚI THIỆU



MỤC TIÊU ĐỀ TÀI

Xây dựng một chương trình mô phỏng việc tự động lái xe thông qua game bằng phương pháp classification.

GIỚI HẠN ĐỀ TÀI

Đề tài dừng lại việc chạy chương trình mô phỏng trong game thông qua ngôn ngữ lập trình python.

PHƯƠNG PHÁP NGHIÊN CỨU

Nghiên cứu các lý thuyết liên quan về chủ đề

ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

Đề tài nghiên cứu các thành phần chính và các lý thuyết liên quan đến hệ thống lái xe trong game. Từ đó cài đặt chương trình và chạy trên máy tính cá nhân bằng ngôn ngữ python



CHƯƠNG 2:

Cơ sở lý thuyết

2.1 Tổng quan về xử lý ảnh

Mục đích của xử lý ảnh

01

Biến đổi làm tăng chất lượng ảnh.

02

Tự động nhận dạng ảnh, đánh giá nội dung ảnh.

03

Kiểm tra, phân loại ảnh.

04

Nhận biết và đánh giá ảnh từ đó có thể phân chia ảnh thành những phần có ý nghĩa để phân biệt giữa đối tượng này với đối tượng khác từ đó áp dụng vào các ứng dụng trong đời sống.

Quá trình xử lý ảnh



2.2 Thư viện OpenCV:

- Là một thư viện phần mềm và thị giác máy tính , mã nguồn mở.

Ứng dụng của OpenCV:

Robot và xe hơi tự lái

Phân tích hình ảnh y học

Tìm kiếm và phục hồi hình ảnh/video

Chức năng của OpenCV:

Xử lý và hiển thị hình ảnh/video

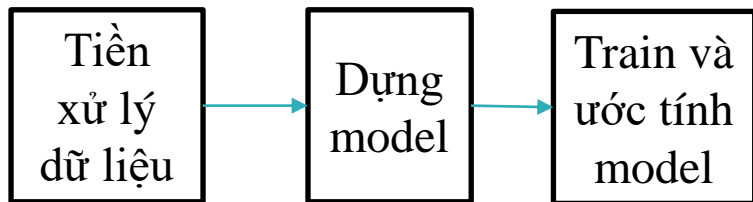
Phát hiện các vật thể

Áp dụng vào học máy và phân cụm



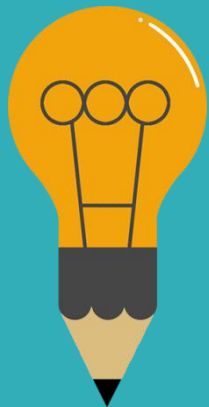
2.3 Thư viện TensorFlow

Cách TensorFlow hoạt động



Lợi ích

- Người dùng có thể dễ dàng truy cập dữ liệu có thể thúc đẩy sự phát triển của AI bằng cách sử dụng Tensorflow.
- Hỗ trợ nhiều ngôn ngữ.
- Xây dựng mạng thần kinh nhân tạo chỉ với vài dòng lệnh.

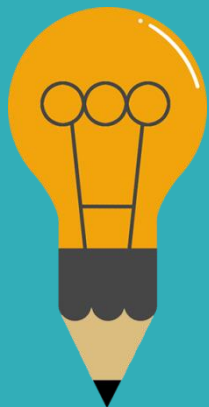


2.4 Thư viện TFlearn

- TFlearn là một thư viện học tập sâu mô-đun và minh bạch được xây dựng trên Tensorflow. Nó được thiết kế để cung cấp API cấp cao hơn cho TensorFlow nhằm tạo điều kiện thuận lợi và tăng tốc các thử nghiệm, trong khi vẫn hoàn toàn minh bạch và tương thích với nó

Ưu điểm:

- TFlearn rất dễ sử dụng và dễ hiểu.
- Nó bao gồm các chức năng trợ giúp mạnh mẽ để đào tạo các bộ tích hợp chấp nhận nhiều đầu vào, đầu ra và trình tối ưu hóa.
- Nó bao gồm trực quan hóa đồ thị dễ dàng và đẹp mắt.

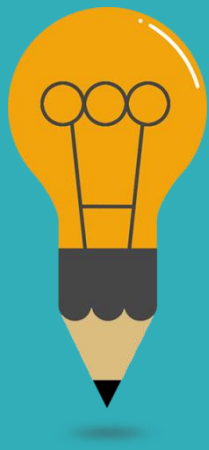
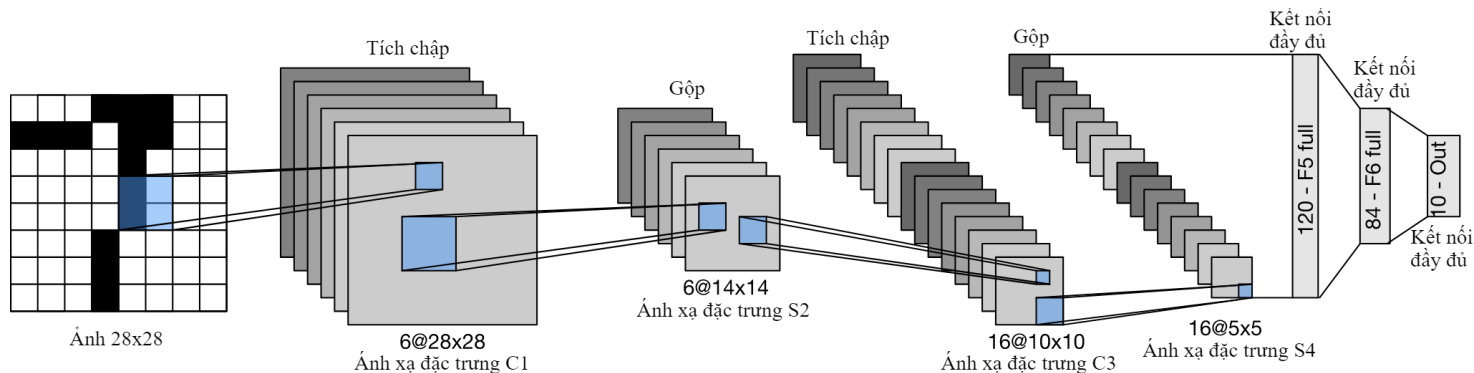


2.5 Mạng Nơ-ron tích chập

Ưu điểm của mạng nơ-ron tích chập

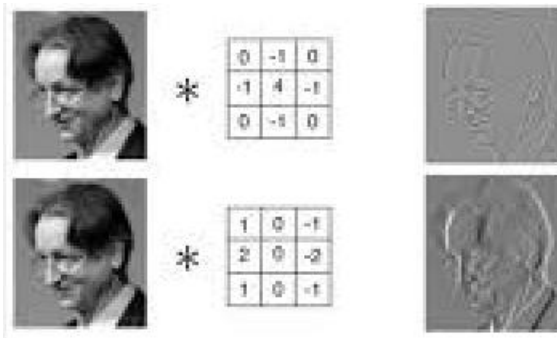
- Các tính năng được mô hình nơ-ron tích chập học một cách trực tiếp.
- Kết quả nhận dạng chính xác cao.
- Có thể học lại, xây dựng các nhiệm vụ nhận dạng mới trên cơ sở các nhiệm vụ nhận dạng cũ.

Kiến trúc mô hình mạng nơ-ron tích chập



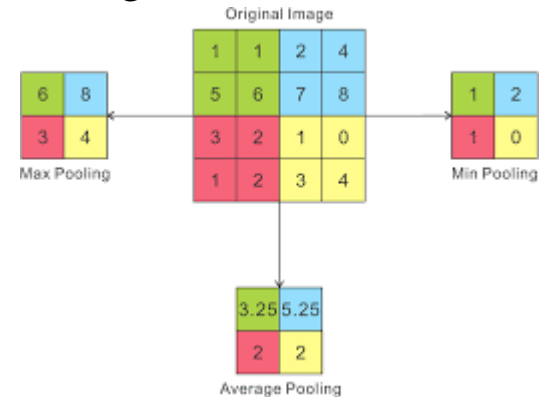
Lớp tích chập (Convolution)

- Lớp quan trọng trong mô hình mạng nơ-ron tích chập tiếp nhận và xử lý hình ảnh
- Khi được áp dụng phép tính tích chập vào xử lý Convolution sẽ giúp biến đổi các thông tin ban đầu vào thành các yếu tố đặc trưng



Lớp gộp (Pooling)

- Lớp này sử dụng một cửa sổ trượt quét qua toàn bộ ảnh dữ liệu, mỗi lần trượt theo một số bước trượt (stride) cho trước.
- Các phương thức lấy phổ biến trong lớp Pooling là Max Pooling (lấy giá trị lớn nhất), Min Pooling (lấy giá trị nhỏ nhất) và Average Pooling (lấy giá trị trung bình)



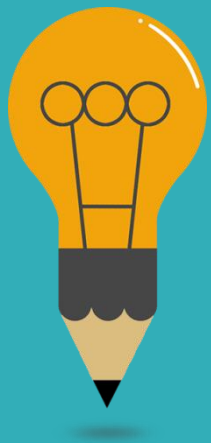


CHƯƠNG 3:

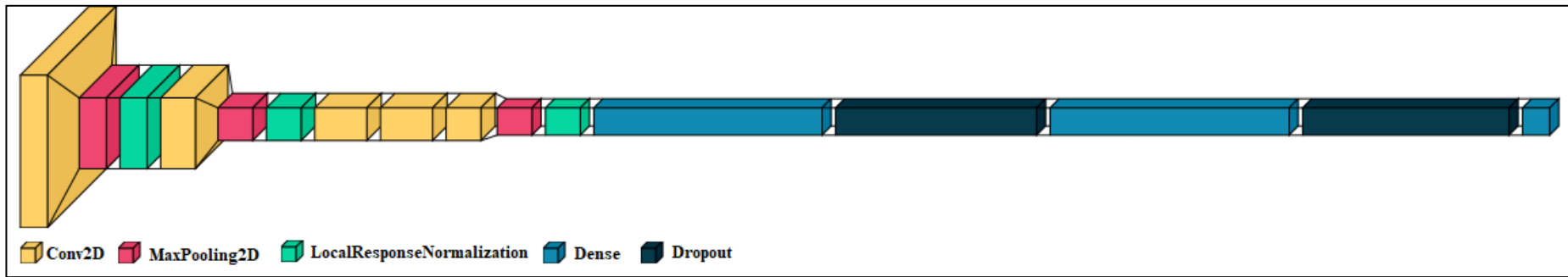
Thiết kế hệ thống

Thiết kế phần mềm

- **Chuẩn bị:** Môi trường, thư viện, thu nhập dữ liệu, cân bằng dữ liệu, directKey.
- **Cơ sở dữ liệu:** Đầu vào dữ liệu được thu nhập bởi hoạt động trong game, tập huấn luyện , tập kiểm tra.
- **Mô hình hóa bài toán:**
 - + **Đầu vào (Input):** Khung hình của tựa game.
 - + **Đầu ra (Output):** Hướng di chuyển của xe .



Mô hình CNN cho hệ thống lái xe tự động trong game:





Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 38, 28, 96)	11712
max_pooling2d (MaxPooling2D)	(None, 18, 13, 96)	0
local_response_normalization (Local Response Normalization)	(None, 18, 13, 96)	384
conv2d_1 (Conv2D)	(None, 18, 13, 256)	614656
max_pooling2d_1 (MaxPooling2D)	(None, 8, 6, 256)	0
local_response_normalization_1 (Local Response Normalization)	(None, 8, 6, 256)	1024
conv2d_2 (Conv2D)	(None, 8, 6, 384)	885120
conv2d_3 (Conv2D)	(None, 8, 6, 384)	1327488
conv2d_4 (Conv2D)	(None, 8, 6, 256)	884992
max_pooling2d_2 (MaxPooling2D)	(None, 3, 2, 256)	0
local_response_normalization_2 (Local Response Normalization)	(None, 3, 2, 256)	1024
dense (Dense)	(None, 3, 2, 4096)	1052672
dropout (Dropout)	(None, 3, 2, 4096)	0
dense_1 (Dense)	(None, 3, 2, 4096)	16781312
dropout_1 (Dropout)	(None, 3, 2, 4096)	0
dense_2 (Dense)	(None, 3, 2, 3)	12291
=====		
Total params: 21,572,675		
Trainable params: 21,571,459		
Non-trainable params: 1,216		

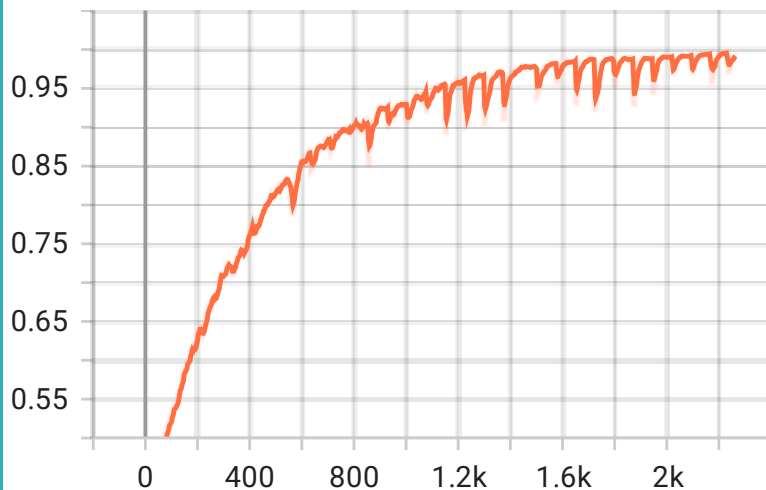


CHƯƠNG 4:

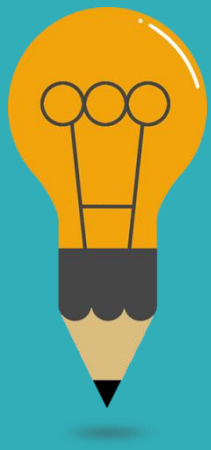
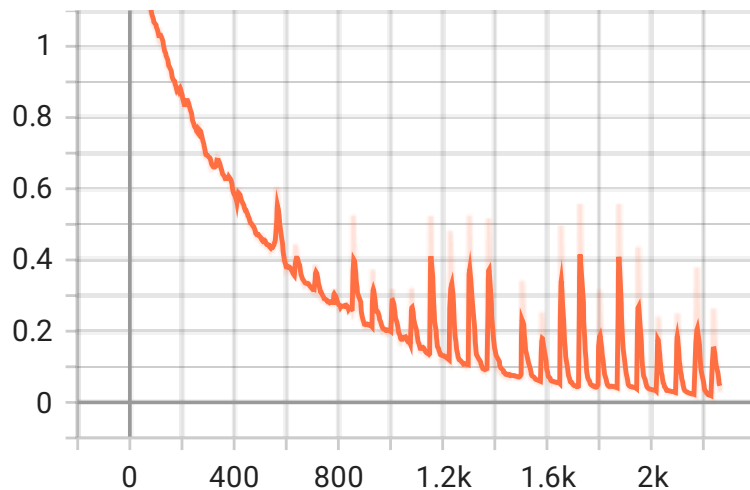
Kết Quả

Thông số trên tập train

Accuracy
tag: Accuracy

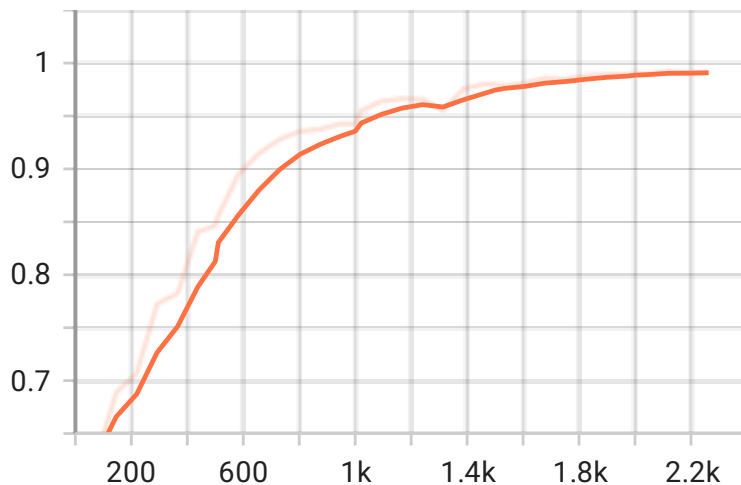


Loss
tag: Loss

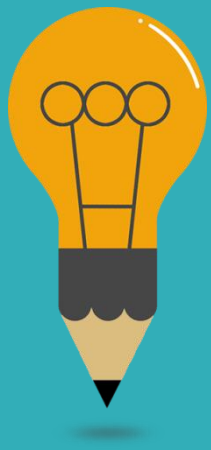
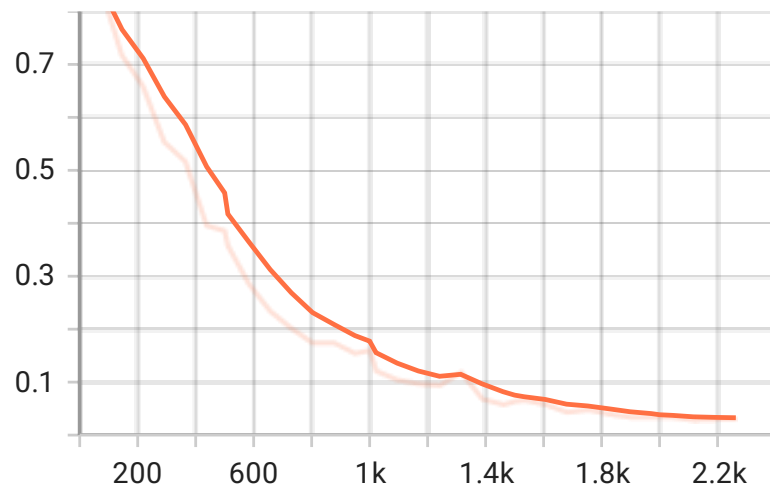


Thông số trên tập test

Accuracy/Validation
tag: Accuracy/Validation



Loss/Validation
tag: Loss/Validation



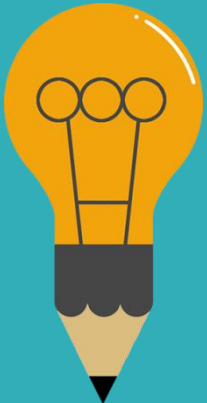


CHƯƠNG 5:

Kết Luận

Kết Luận

- Về khả năng lái xe , hệ thống thực hiện khá chính xác đối với những nơi không có người và xe cộ. Đối với những nơi đông đúc xe cộ và người thì hệ thống hoạt động ở mức ổn. Hệ thống vẫn còn chưa thực sự hiệu quả vì model không có nhận diện vật thể do đó model không biết dừng khi đèn đỏ hay né những chướng ngại vật bất ngờ .
- Cần giải quyết những nhược điểm mà phần mềm còn đang gặp phải, thực hiện thêm ý tưởng mới nhằm nâng cao độ hiệu quả , hiệu suất và độ chính xác của chương trình.





Thank you