

	VIETTEL AI RACE	TD084
	Hệ Thống Thị Giác Máy Tính Cho Xe Tự Hành Thể Hệ Tiếp Theo	Lần ban hành: 1

1. Giới thiệu

Xe tự hành đang trở thành tâm điểm của ngành giao thông thông minh. Để hoạt động an toàn trong môi trường phức tạp, phương tiện phải nhận thức xung quanh nhanh, chính xác và tin cậy. Trọng tâm của khả năng này là thị giác máy tính (computer vision) – công nghệ giúp xe “nhìn” và “hiểu” thế giới thông qua camera, radar, LiDAR và cảm biến đa dạng.

Thể hệ tiếp theo của xe tự hành yêu cầu hệ thống thị giác máy tính vượt xa các mô hình hiện tại, hỗ trợ nhận diện thời gian thực, dự đoán hành vi, và ra quyết định tự động trong điều kiện thời tiết, ánh sáng và giao thông phức tạp.

2. Cấu phần chính của hệ thống thị giác

- Cảm biến đa phương thức:
 - Camera RGB/IR: Thu thập hình ảnh màu và hồng ngoại, xử lý chi tiết môi trường.
 - LiDAR: Tạo bản đồ 3D chính xác của môi trường, đo khoảng cách và hình dạng vật thể.
 - Radar sóng mm: Hoạt động tốt trong sương mù, mưa, tuyết, bỗ trợ cho camera.
 - Cảm biến siêu âm: Hỗ trợ đỗ xe, đo khoảng cách gần.
- Xử lý cảm biến hợp nhất (Sensor Fusion):
 - Kết hợp dữ liệu từ nhiều cảm biến để có bức tranh toàn cảnh, giảm sai số từng loại cảm biến.
- Mô-đun thị giác máy tính:
 - Phát hiện đối tượng (Object Detection): Nhận diện người đi bộ, xe cộ, biển báo.
 - Phân đoạn ngữ nghĩa (Semantic Segmentation): Phân loại từng điểm ảnh để xác định làn đường, vỉa hè.
 - Theo dõi chuyển động (Object Tracking): Dự đoán quỹ đạo vật thể di chuyển.

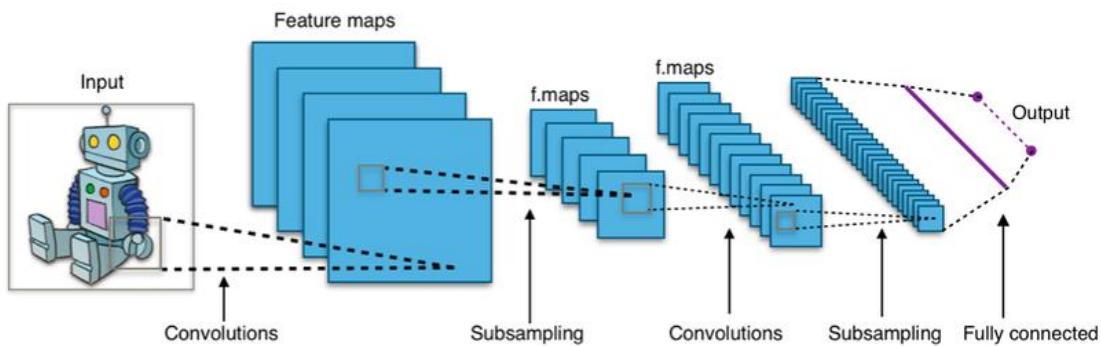
3. Thuật toán và mô hình AI

3.1 Mạng CNN

- Mạng CNN là một tập hợp các lớp Convolution chồng lên nhau và sử dụng các hàm nonlinear activation như ReLU và tanh để kích hoạt các trọng số trong các node. Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo.

	VIETTEL AI RACE Hệ Thống Thị Giác Máy Tính Cho Xe Tự Hành Thế Hệ Tiếp Theo	TD084
		Lần ban hành: 1

- Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo. Trong mô hình mạng truyền ngược (feedforward neural network) thì mỗi neural đầu vào (input node) cho mỗi neural đầu ra trong các lớp tiếp theo.
- Mô hình này gọi là mạng kết nối đầy đủ (fully connected layer) hay mạng toàn vẹn (affine layer). Còn trong mô hình CNNs thì ngược lại. Các layer liên kết được với nhau thông qua cơ chế convolution.
- Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Như vậy mỗi neuron ở lớp kế tiếp sinh ra từ kết quả của filter áp đặt lên một vùng ảnh cục bộ của neuron trước đó.
- Mỗi một lớp được sử dụng các filter khác nhau thông thường có hàng trăm hàng nghìn filter như vậy và kết hợp kết quả của chúng lại. Ngoài ra có một số layer khác như pooling/subsampling layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu).
- Trong quá trình huấn luyện mạng (traning) CNN tự động học các giá trị qua các lớp filter dựa vào cách thức mà bạn thực hiện. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự raw pixel > edges > shapes > facial > high-level features. Layer cuối cùng được dùng để phân lớp ảnh.

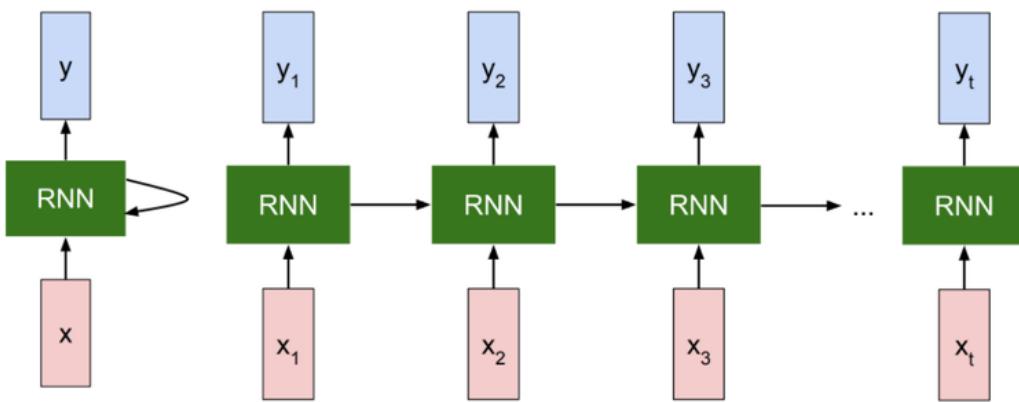


-
- Trong mô hình CNN có 2 khía cạnh cần quan tâm là **tính bát biến** (Location Invariance) và **tính kết hợp** (Compositionality). Với cùng một đối tượng, nếu đổi tượng này được chiếu theo các góc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể.
- Pooling layer sẽ cho bạn tính bát biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling). Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các filter.
- Đó là lý do tại sao CNNs cho ra mô hình với độ chính xác rất cao. Cũng giống như cách con người nhận biết các vật thể trong tự nhiên.

	VIETTEL AI RACE Hệ Thống Thị Giác Máy Tính Cho Xe Tự Hành Thế Hệ Tiếp Theo	TD084
		Lần ban hành: 1

- Mạng CNN sử dụng 3 ý tưởng cơ bản:
- **các trường tiếp nhận cục bộ** (local receptive field)
- **trọng số chia sẻ** (shared weights)
- **tổng hợp** (pooling).

3.2 Recurrent Neural Network



- Các x ở đây đại diện cho dữ liệu đầu vào lần lượt (được chia theo time step).
- x_{txt} đại diện cho time step thứ t , và y_{tyt} là output của một step. Ví dụ, x_2x_2 sẽ là vector đại diện của từ thứ 2 trong câu văn bản.
- Hình ảnh dưới đây cho thấy rõ hơn điều gì thực sự xảy ra trong một step.
-
- **Hidden state h_{ht}** (trong một số tài liệu thường ký hiệu s_{st}). Đây chính là bộ nhớ của mạng. h_{ht} là tổng hợp thông tin của hidden state trước (h_{t-1}) cộng với input tại time step t (x_{xt}). Activation function ở đây \$g_1\$ chủ yếu là tanh hoặc ReLu.
- $h_t = g_1(W_{hh} \cdot h_{t-1} + W_{hx} \cdot x_t + b_h)$
- Hoặc có thể viết gọn hơn:
- $h_t = g_1((W_{hh}W_{hx})(h_{t-1}x_t))$
- $h_t = g_1((W)(h_{t-1}x_t))$
- **Output của từng time step y_{tyt}** : Tại 1 block của mạng RNN có 2 đầu ra. Trong đó, h_{ht} là tổng hợp thông tin của các state trước để tiếp tục truyền đi trong chuỗi mạng, và ta có thêm y_{tyt} là output của từng time step một. Ở đây g_2 thường là hàm softmax.
- $y_t = g_2(W_{yh} \cdot h_t + b_y)$

	VIETTEL AI RACE	TD084
	Hệ Thống Thị Giác Máy Tính Cho Xe Tự Hành Thể Hệ Tiếp Theo	Lần ban hành: 1

4. Ứng dụng và kịch bản thực tế

- Điều khiển trên đường cao tốc: Nhận dạng phương tiện di chuyển nhanh, dự đoán làn đường.
- Môi trường đô thị đông đúc: Xử lý tình huống bất ngờ như người đi bộ băng qua đường, xe đạp di chuyển ngược chiều.
- Thời tiết khắc nghiệt: Cần thuật toán chịu nhiều tốt trong mưa lớn, tuyết dày, hoặc ánh sáng yếu.
- Đỗ xe tự động: Nhận diện vật cản nhỏ, khoảng trống, biển báo trong không gian hẹp.

5. Thách thức kỹ thuật

- Đa dạng môi trường: Khác biệt về điều kiện ánh sáng, thời tiết, hạ tầng đường xá giữa các quốc gia.
- Xử lý thời gian thực: Đòi hỏi phần cứng GPU/TPU mạnh, tối ưu hóa mô hình để giảm độ trễ dưới 50 ms.
- An toàn và tin cậy: Hệ thống phải có cơ chế dự phòng khi một hoặc nhiều cảm biến hỏng.
- Dữ liệu huấn luyện khổng lồ: Cần hàng triệu km dữ liệu lái xe, bao gồm tình huống hiếm gặp.
- Tấn công đối kháng (Adversarial Attacks): Cần bảo vệ trước việc kẻ xấu dùng biển báo giả hay tín hiệu gây nhiễu.

6. Xu hướng phát triển

- Edge AI trong xe: Xử lý trực tiếp trên thiết bị thay vì gửi về đám mây, giảm độ trễ và tăng tính riêng tư.
- LiDAR thế hệ mới chi phí thấp: Giúp phổ cập xe tự hành.
- Mô hình học tự giám sát (Self-Supervised Learning): Giảm nhu cầu dữ liệu gắn nhãn vốn kén.
- Hợp tác xe – hạ tầng (V2X): Xe trao đổi dữ liệu trực tiếp với đèn tín hiệu, camera giao thông, và các phương tiện khác.
- Giải thích mô hình (Explainable AI): Cho phép hiểu quyết định của hệ thống để hỗ trợ kiểm định an toàn.

7. Tác động xã hội và kinh tế

- An toàn giao thông: Giảm tai nạn do lỗi con người.
- Hiệu quả vận tải: Giảm ùn tắc, tối ưu hóa lộ trình và tiết kiệm nhiên liệu.

	VIETTEL AI RACE	TD084
	Hệ Thống Thị Giác Máy Tính Cho Xe Tự Hành Thể Hệ Tiếp Theo	Lần ban hành: 1

- Dịch vụ di chuyển mới: Robotaxi, giao hàng tự động, xe buýt tự lái.
- Quy định pháp lý: Cần khung pháp luật cho bảo hiểm, trách nhiệm tai nạn và quyền riêng tư.