Báo cáo luận văn

Tên luận văn: Xe tự lái

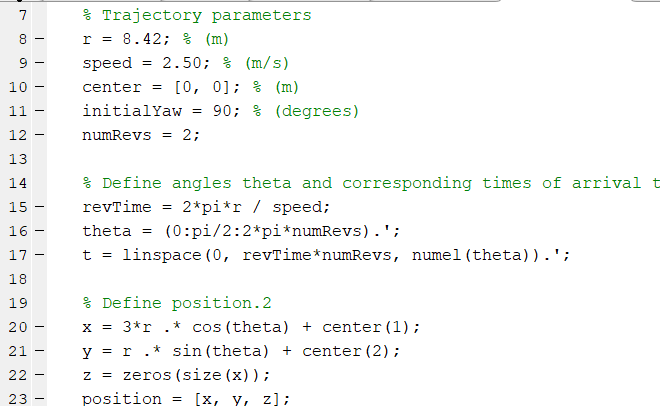
Nhiệm vụ mong muốn: Xe tự lái có khả năng định vị vị trí sử dụng IMU/GPS/Encoder

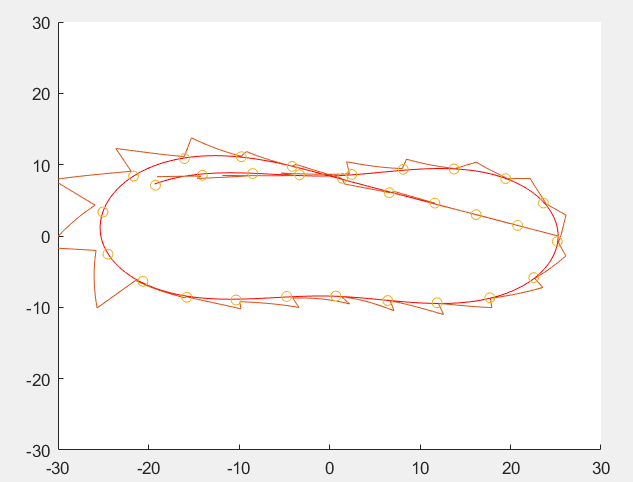
Xe tự lái có khả năng chạy trên đường nhựa, giữ làn đường

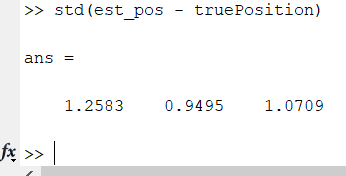
Xe tự lái có khả năng lên kế hoạch đường đi dựa vào Google Map API

Kết quả thực hiện:

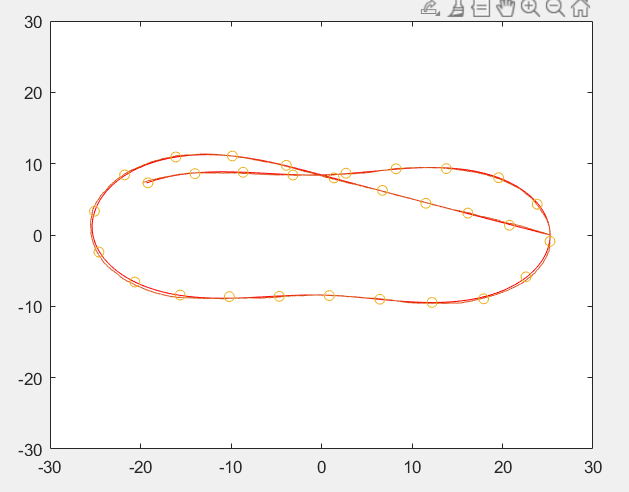
Nhiệm vụ 1: Mô phỏng thuật toán Error State Extended Kalman Filter để kết hợp dữ liệu thu được từ IMU và GPS

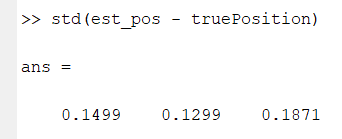






Bước tiếp theo: Kết hợp dữ liệu từ Encoder và camera để tăng độ chính xác định vị

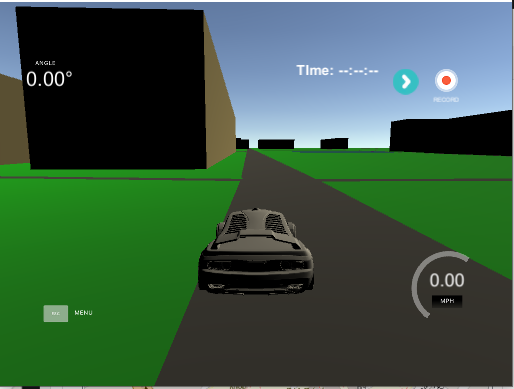




Nhiệm vụ tiếp theo: Viết lại code cho bộ lọc này trên C++, Đã thực hiện

Nhiệm vụ 2: Giữ làm đường, chạy trên làn đường:

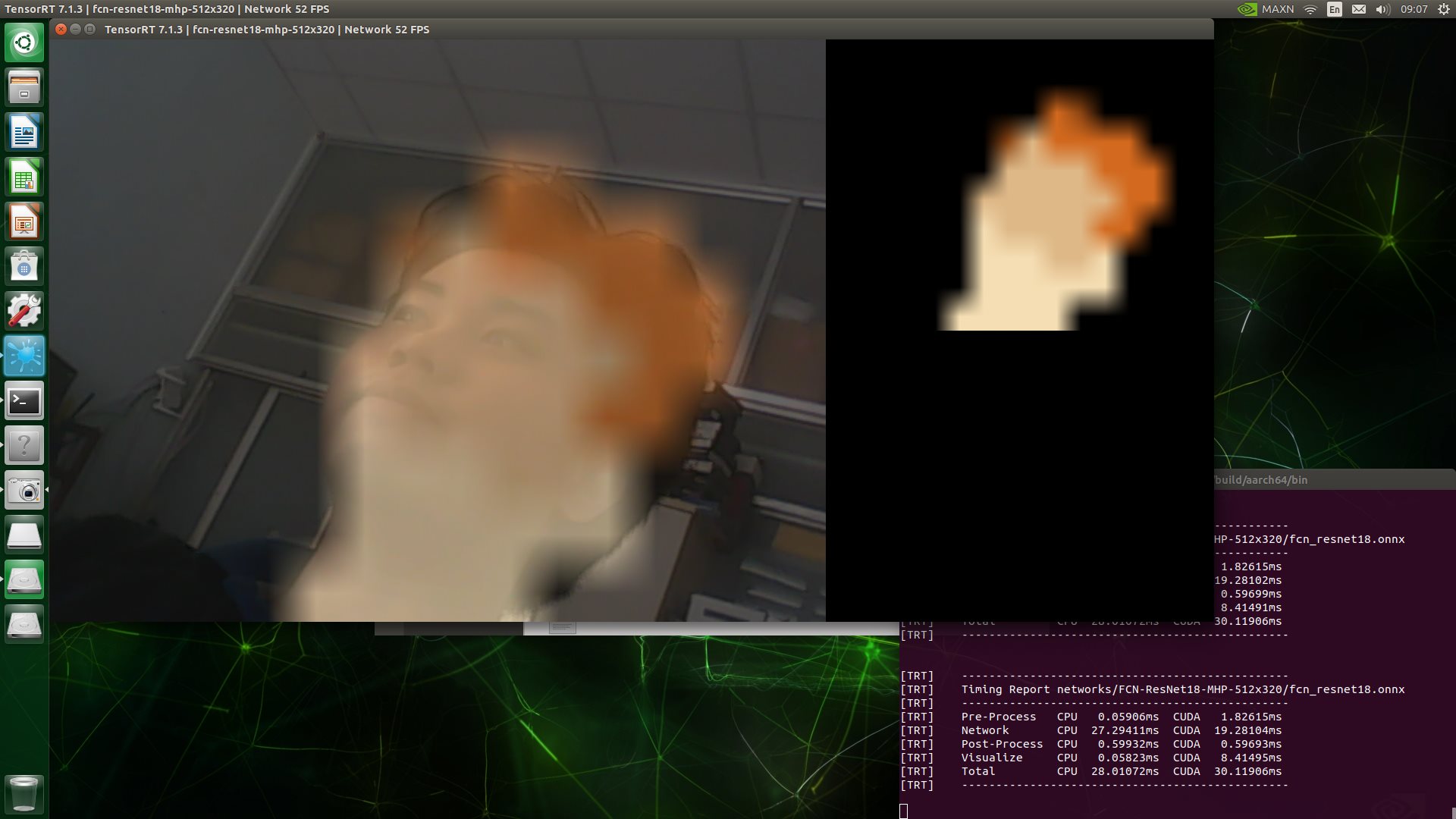
Mô phỏng xe trên Unity:



Mô phỏng được thực hiện dựa trên bản đồ trường Bách Khoa HCM, được xuất qua file obj để import vào map, mô phỏng sẽ truyền tín hiệu vị trí, tín hiệu mô phỏng GPS, vận tốc, giá trị đo của IMU và ảnh camera Bird Eye View và Camera hướng thẳng về GUI bằng giao thức ZMQ.

Nhiệm vụ: Sử dụng phương pháp Sematic Segmentation tạo ra Ocuppcy Map. Từ đó sử dụng thuật toán tìm đường để lên kế hoạch xe chạy

Kết quả thực hiện: Test model segmentation trên Jetson Tx1: <https://github.com/dusty-nv/jetson-inference/blob/master/docs/segnet-console-2.md>



Kết quả: Model chạy trên Jetson Tx1 đạt được tầm 48 – 50fps cho các model fcn-resnet18-cityscapes-1024x512 và fcn-resnet18-deepscene-576x320

Tuy nhiên độ chính xác chưa đảm bảo, ảnh segmentation bị vỡ khi di chuyển nhanh.

Nhiệm vụ 3: Tìm đường trên map + Điều khiển:

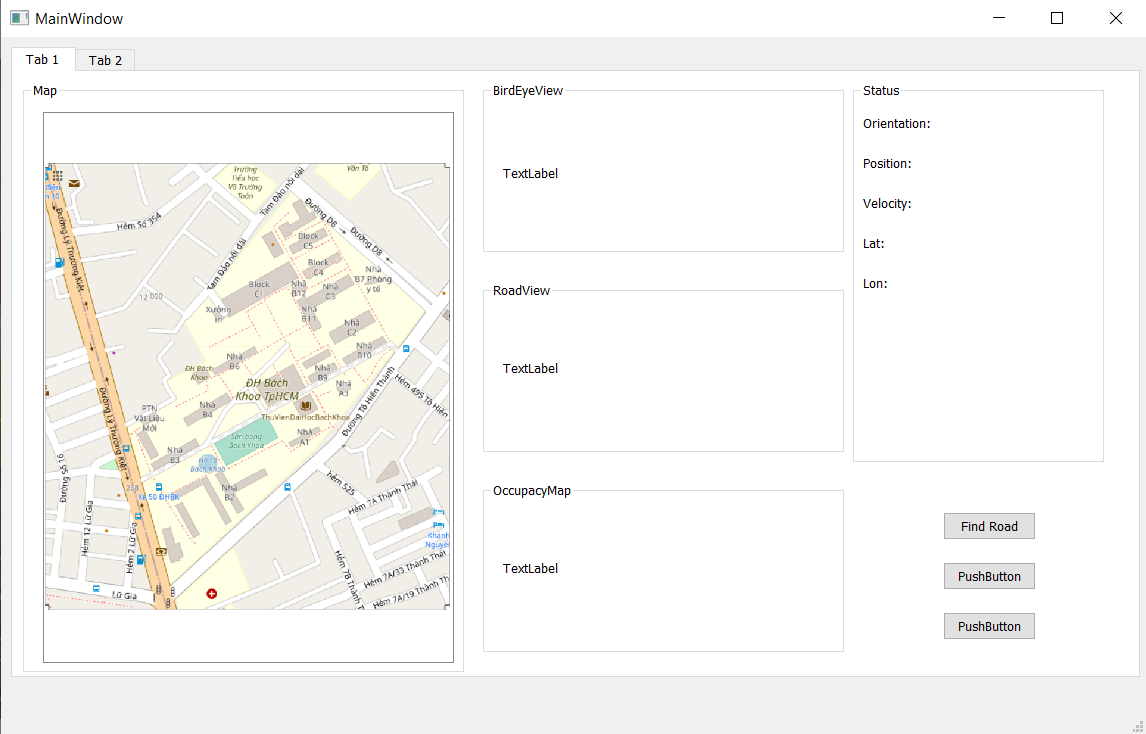
Sử dụng Google Map API để tìm ra vị trí cần đến, đưa ra chỉ thị cho lớp dưới thực thi (Waypoint mission) để tạo đường đi toàn cục (global planner)

Ở quy mô nhỏ hơn, em sử dụng data từ phần segmentation + data vị trí để đưa ra vị trí dựa trên thuật toán A\*

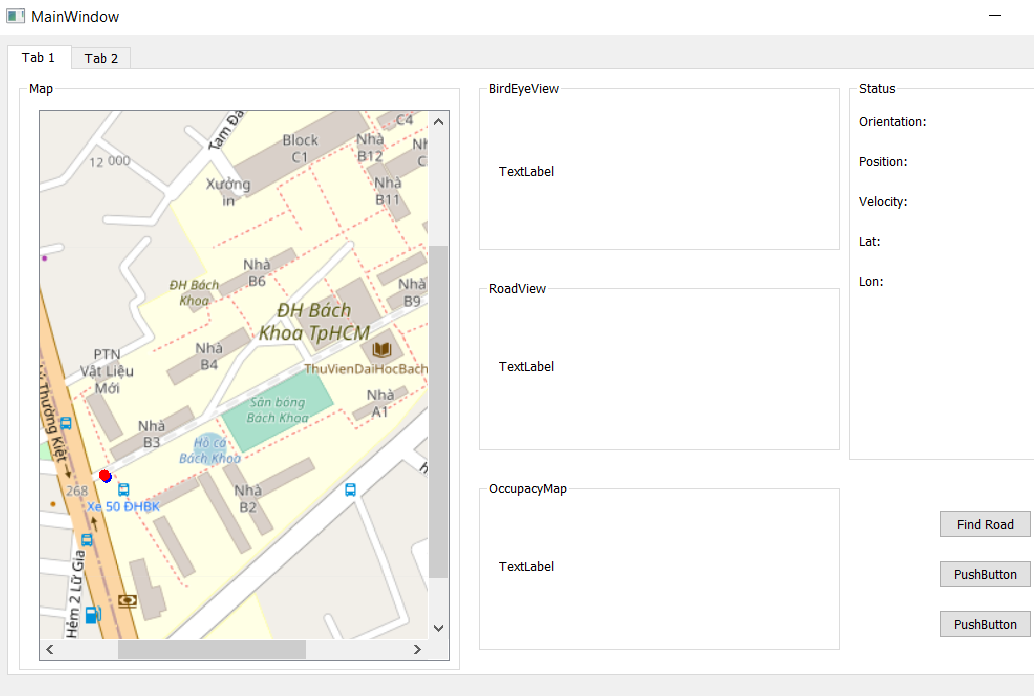
### Phương pháp dự định

Sử dụng map OpenStreetMap để tìm các điểm cần tới, gọi là bước global planning, ở quy mô nhỏ, xe sử dụng Occupancy map đã thu được ở trên kết hợp với thuật toán tìm đường A\* để tìm đường đi cho xe.

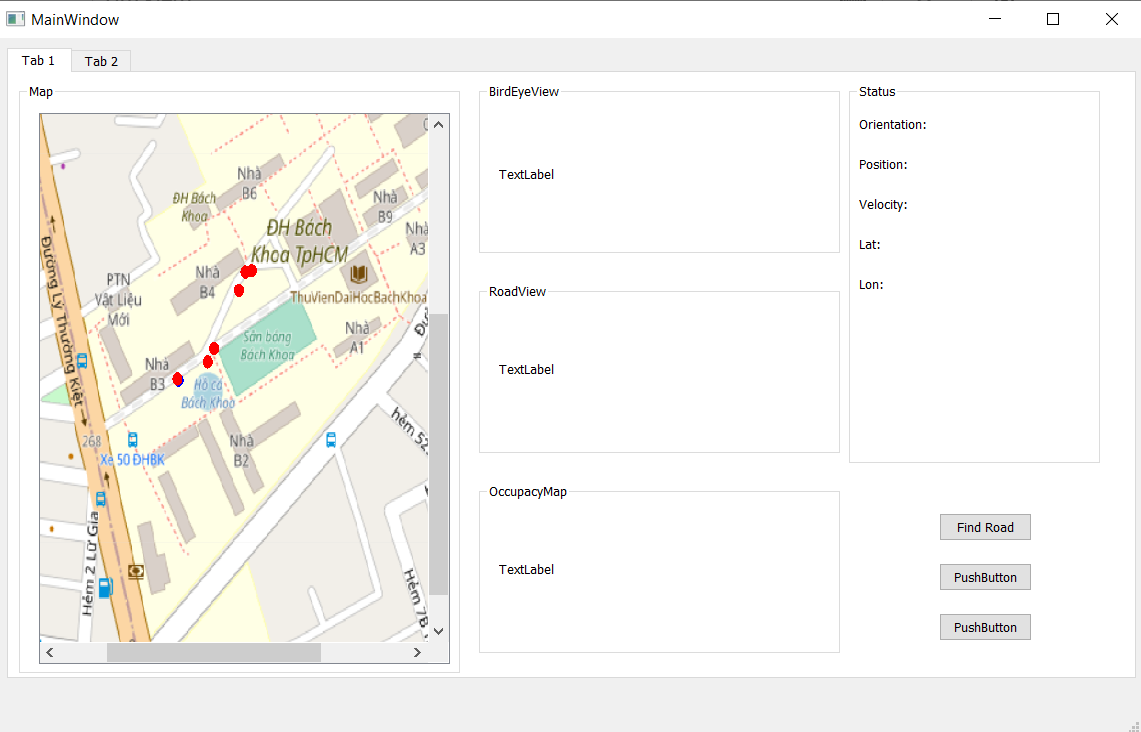
Tiến độ thực hiện: Xây dựng GUI hiển thị map, hiện tại sử dụng OpenStreetMap API:



Thuật toán A\* tìm đường trên map:

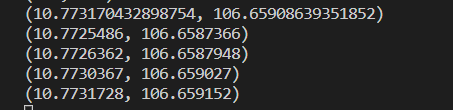


Thu phóng map, chọn điểm đầu – cuối

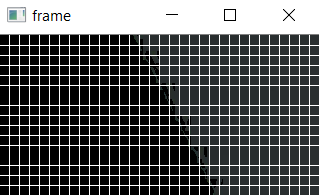
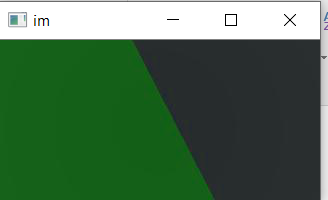


Tìm đường đi cho xe.

Các tọa độ checkpoint như sau:



Ở quy mô cục bộ:

Hình ảnh được đưa về khung hình bird eye view và sau khi được segmentation sẽ được chia thành lưới dựa trên kích thước của ảnh và thông số calibration để xác định khoảng cách và vật cản.  




Dựa trên thông tin từ map này, em sử dụng thuật toán A\* như ở đây để tìm đường đi, với hàm Heuristics là khoảng cách từ vị trí hiện tại của xe cho đến waypoint tiếp theo trên đường đi toàn cục:

<https://www.redblobgames.com/pathfinding/a-star/implementation.html>

Ngoài ra em còn thay đổi một chút ở hàm heuristics ở chỗ khi xe gặp vật cản, nó sẽ tính toán đường đi sao cho cách vật cản 1 đoạn nhất định.

Phần cứng:

Hiện tại phần cứng sử dụng các thành phần sau:

Module ngoại vi kết nối qua UART