TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH

MÔN HỌC

CS331 - THỊ GIÁC MÁY TÍNH NÂNG CAO

BÁO CÁO ĐỒ ÁN
THEO DÕI VÀ ƯỚC TÍNH KHOẢNG CÁCH ĐỐI TƯỢNG TRONG XE TỰ LÁI

Giảng Viên hướng dẫn: Thầy Mai Tiến Dũng

Nhóm:

Bùi Mạnh Hùng - 21522110

TP.HCM, Ngày 20 tháng 12 năm 2023

MỤC LỤC

1 GIỚI THIỆU

1.1 Động lực

Điều quan trọng đối với xe tự lái là khả năng theo dõi và ước tính khoảng cách đến các đối tượng xung quanh. Nhiệm vụ này không chỉ mang lại sự an toàn bằng cách giúp xe tránh va chạm với những đối tượng xung quanh, mà còn nâng cao khả năng lái xe bằng cách thực hiện các thao tác lái xe phức tạp như chuyển làn và vượt xe. Đồng thời, khả năng này cũng giúp xe có thể lái xe hiệu quả trong các điều kiện môi trường khó khăn như đường đông đúc hoặc đường tối.

1.2 Định nghĩa bài toán

Bài toán hiện tại tập trung vào việc nhận dạng và theo dõi các đối tượng trong môi trường xung quanh xe tự lái. Đối tượng ở đây là bất kỳ thực thể nào có thể được nhận biết, ví dụ như xe tải, người đi bộ, người đi xe đạp, và các thực thể khác có thể xuất hiện trong tầm nhìn của hệ thống cảm biến.

Quá trình theo dõi trong bài toán này đòi hỏi khả năng xác định vị trí của các đối tượng này trong thời gian thực. Điều này đòi hỏi hệ thống phải liên tục theo dõi và cập nhật vị trí của các đối tượng trong khi chúng di chuyển trong không gian xung quanh xe tự lái.

Ngoài ra, bài toán còn liên quan đến việc ước tính khoảng cách giữa xe và các đối tượng xung quanh. Cụ thể, quá trình này bao gồm việc xác định khoảng cách từ camera hoặc các thiết bị cảm biến gắn trên xe tự lái đến các đối tượng được nhận dạng. Chức năng này không chỉ đảm bảo an toàn cho xe và những người tham gia giao thông mà còn cung cấp thông tin quan trọng để hỗ trợ quyết định lái xe, chẳng hạn như tránh va chạm và điều chỉnh hành vi lái xe theo cách thông minh và hiệu quả.

1.3 Thách thức

Bài toán đặt ra một số thách thức đối với hệ thống xe tự lái:

- **Môi trường phức tạp:** Môi trường xung quanh xe tự lái thường đầy đủ với nhiều đối tượng khác nhau, có kích thước, hình dạng và màu sắc đa dạng. Điều này làm tăng độ phức tạp của quá trình nhận dạng và theo dõi các đối tượng trong môi trường.
- **Thay đổi nhanh chóng:** Các đối tượng xung quanh có khả năng di chuyển nhanh chóng, đặt ra thách thức đối với quá trình theo dõi và ước tính khoảng cách. Sự động đậy và thay đổi vị trí của các đối tượng đòi hỏi hệ thống phải có khả năng xử lý dữ liệu với tốc độ cao và đưa ra quyết định nhanh chóng.
- **Tác động của môi trường:** Các yếu tố môi trường như ánh sáng và thời tiết có thể ảnh hưởng đáng kể đến khả năng nhận dạng và ước tính khoảng cách đến các đối tượng. Điều

này đặt ra thách thức trong việc duy trì hiệu suất của hệ thống trong mọi điều kiện môi trường.

- **Dữ liệu hạn chế:** Việc thu thập dữ liệu về khoảng cách giữa xe tự lái và các đối tượng xung quanh đang gặp khó khăn. Sự thiếu hụt về dữ liệu đồng nghĩa với việc hệ thống cần có khả năng tự đào tạo và làm việc hiệu quả trên các tình huống mới mà nó có thể gặp phải.

2 Phương pháp thực hiện

2.1 Tổng quan phương pháp

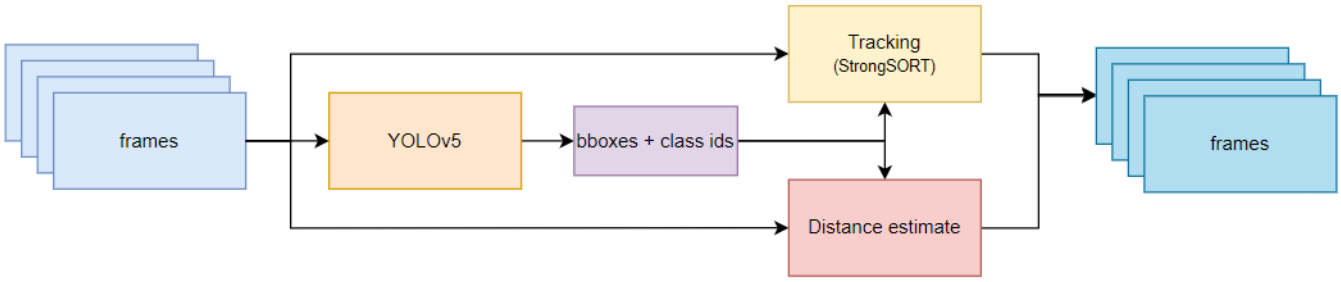

Figure 1: Tổng quan phương pháp sử dụng

Mô tả tổng quan phương pháp: Hệ thống nhận đầu vào là một chuỗi các khung hình, sau đó lần lượt qua mô hình YOLO để phát hiện các đối tượng có trong khung hình. Trong phương pháp đề xuất, nhóm sử dụng thuật toán theo dõi đối tượng là strongSORT là một thuật toán được cải tiến từ deepSORT, ngoài ra nhóm sử dụng mô hình GLPDepth để ước tính độ sâu khung hình đồng thời kết hợp với kết quả detection từ YOLO để xây dựng mô hình ước tính khoảng cách đến các đối tượng, dưới đây là trình bày 3 mô-đun chính của phương pháp :

1. **Phát hiện đối tượng với YOLOv5**
   - Trong bước này, mỗi khung hình được đưa qua mô hình YOLOv5 để phát hiện các đối tượng trong ảnh.
   - Kết quả đầu ra bao gồm hộp giới hạn (bbox) và lớp (class id) tương ứng với mỗi đối tượng được phát hiện.
2. **Kết hợp dự đoán từ YOLOv5 với thuật toán theo dõi đối tượng**
   - **Module Tracking:** Thực hiện nhiệm vụ theo dõi, giúp theo dõi chuyển động của các đối tượng qua các khung hình liên tiếp.
3. **Kết hợp dự đoán từ YOLOv5 với thuật toán ước tính khoảng cách**

- **Module Distance Estimate:** Thực hiện nhiệm vụ ước tính khoảng cách từ camera hoặc các thiết bị cảm biến đến từng đối tượng, cung cấp thông tin về vị trí không gian của chúng.

Kết quả cuối cùng là tổng hợp các thông tin chi tiết về đối tượng như vị trí (bbox), loại đối tượng (class id), thông tin theo dõi chuyển động (ID) và ước tính khoảng cách. Hệ thống này giúp cải thiện khả năng phát hiện, theo dõi và đánh giá khoảng cách đối với các đối tượng trong môi trường xung quanh.

2.2 Phương pháp phát hiện đối tượng

Phương pháp:

- YOLOv5 là phương pháp phát hiện đối tượng thuộc họ phương pháp YOLO, và thuộc loại Onestage detector, hình ảnh đầu vào sẽ đi vào mô hình chỉ một lần để trả về kết quả là bounding box và kết quả lớp đối tượng.
- YOLOv5 sử dụng kiến trúc mạng nơ-ron tích chập để phát hiện đối tượng. Nó chia ảnh thành lưới và dự đoán các hộp giới hạn chô mô ô trong lưới
- Mô hình được huấn luyện trên tập dữ liệu COCO và đạt được độ chính xác cao trong phát hiện đối tượng thời gian thực

Input:

- YOLOv5 nhận đầu vào là hình ảnh

Output:

- Kết quả đầu ra của YOLOv5 bao gồm bounding box (hộp giới hạn) và Class ID của mỗi đối tượng được phát hiện trong ảnh.
  - Bounding box cung cấp thông tin về vị trí và kích thước của đối tượng trên hình ảnh.
  - Class ID xác định loại đối tượng cụ thể mà mô hình đã nhận diện.

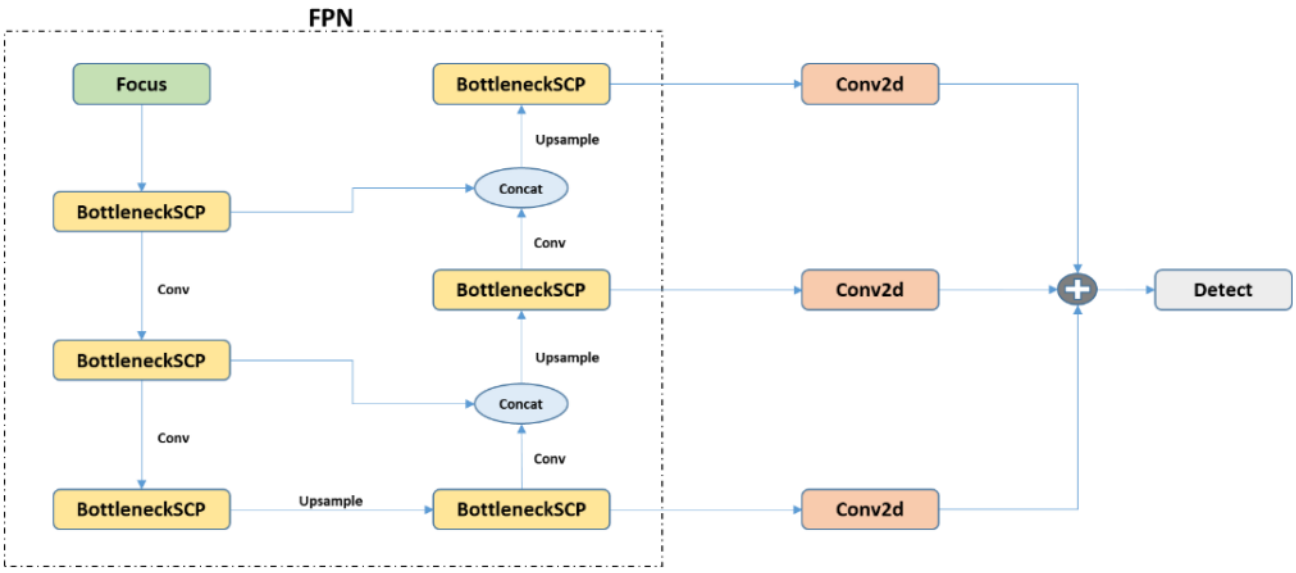

Figure 2: Kiến trúc mô hình YOLOv5

2.3 Phương pháp theo dõi đối tượng

Tổng Quan:

- StrongSORT (Strongly-Correlated Multiple Object Tracking) là một phương pháp theo dõi đối tượng hiệu quả trong các chuỗi video.
- Mục tiêu chính của StrongSORT là gán một ID duy nhất cho mỗi đối tượng và theo dõi chúng qua các frame của video mà không bị mất mát hoặc nhầm lẫn.

Phương Pháp:

- StrongSORT là một phương pháp theo dõi đối tượng dựa trên sự kết hợp mạng học sâu (deep neural networks) và kỹ thuật thị giác máy tính truyền thống. StrongSORT được thiết kế để giải quyết một số hạn chế của thuật toán theo dõi đối tượng trước đây (SORT, deepSORT,..) đồng thời tăng độ chính xác trong việc theo dõi đối tượng thông qua các khung hình
- Một trong những kỹ thuật chính của strongSort là sử dụng mạng học sâu để trích xuất đặc trưng từ hình ảnh để cải thiện khả năng theo dõi trước sự thay đổi về độ sáng , tầm nhìn, bị che khuất của input đầu vào. StrongSORT sử dụng bộ lọc Kalman để mô hình hóa chuyển động của đối tượng mục tiêu and cũng như ước tính vị trí của nó trong chuỗi các khung hình.
- Mô hình này sử dụng dữ liệu đầu vào là các bounding box được phát hiện từ YOLOv5 thông qua mỗi frame của video.

Input:

- **Video Frames:** Chuỗi các khung hình từ video, đưa vào hệ thống để thực hiện quá trình theo dõi.
- **Bounding Box từ YOLOv5:** Các bounding box chứa thông tin về vị trí và kích thước của đối tượng được phát hiện trong mỗi frame.

Output:

- **ID và Vị Trí Bounding Box:** Kết quả đầu ra của StrongSORT bao gồm ID duy nhất được gán cho mỗi đối tượng và vị trí của bounding box của đối tượng đó trong từng frame của video.

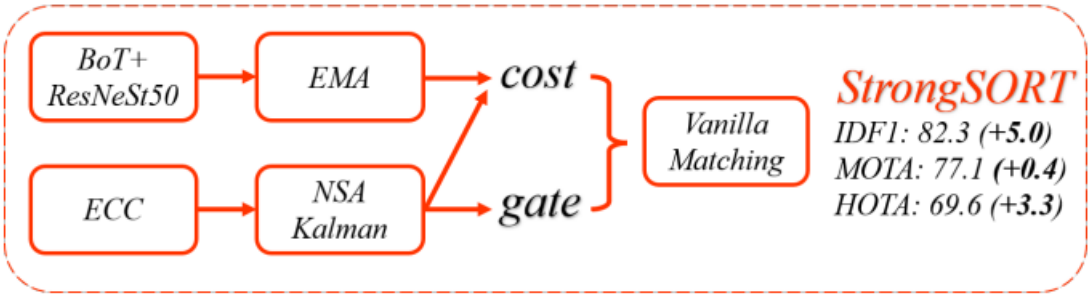

Figure 3: Kiến trúc mô hình StrongSORT

2.4 Phương pháp ước tính khoảng cách

Phương Pháp:

- Mô hình GLPDepth + LSTM kết hợp giữa mô hình ước tính độ sâu GLPDepth và mạng neural dài hạn ngắn hạn (LSTM) để dự đoán khoảng cách từ camera gắn trên xe đến mỗi đối tượng được phát hiện trong ảnh.

Input:

- **Frame + Bboxes + Class IDs:**
  - **Frame:** Hình ảnh đầu vào.
  - **Bounding Boxes (Bboxes):** Thông tin về vị trí và kích thước của các bounding box chứa đối tượng.
  - **Class IDs:** Loại của từng đối tượng được xác định từ bounding box.

Output:

- **Khoảng Cách:** Kết quả đầu ra của mô hình là ước tính khoảng cách từ camera gắn trên xe đến mỗi đối tượng được phát hiện trong hình ảnh.

Chi Tiết Về Mô Hình:

1. **Ước Tính Độ Sâu (GLPDepth):**

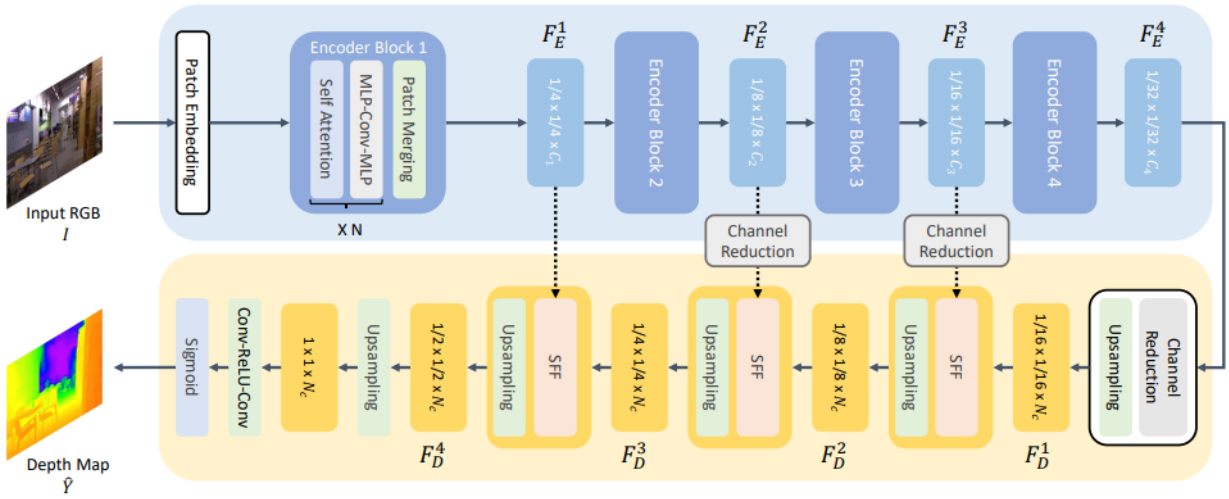

Figure 4: Tổng quan kiến trúc GLPDepth. Những thành phần chính của kiến trúc là encoder, decoder, và skip-connections với các mô-đun tổng hợp đặc trưng

Phương pháp

- Phương pháp sử dụng kiến trúc Global-Local Path Networks gồm 2 nhánh. Nhánh Global để nắm bắt các tính chất toàn cục của ảnh. Nhánh Local để chi tiết hóa ở khu vực quan tâm.
- GLPDepth sử dụng phương pháp Vertical CutDepth để chia ảnh ra các khu vực theo chiều dọc, giúp nhánh Local tập trung vào từng vùng riêng biệt

Input: Hình ảnh đơn 2D
Output: Bản đồ độ sâu (depth map) là một ma trận có cùng kích thước với ảnh input, mỗi phần tử trong ma trận chứa giá trị ước tính độ sâu của phần tử ảnh tương ứng

Mô hình GLPDepth được sử dụng trong bài toán này để từ hình ảnh đầu vào, ước tính độ sâu để tạo ra bản đồ độ sâu (depth map). Điều này giúp mô hình ước tính khoảng cách có cái nhìn về cấu trúc không gian của đối tượng trong ảnh.

2. **Ánh Xạ Tọa Độ và Class IDs:** Bản đồ độ sâu sau đó được sử dụng để ánh xạ tọa độ và class IDs của mỗi bounding box lên bản đồ độ sâu. Việc này giúp trích xuất các đặc trưng cần thiết cho việc dự đoán khoảng cách.

3. **Mô Hình LSTM:**

    - Đầu vào của mô hình LSTM là các đặc trưng đã được trích xuất từ bản đồ độ sâu và thông tin khác như tọa độ và loại của từng đối tượng.
    - LSTM được sử dụng để học và hiểu các mối quan hệ không gian và thời gian giữa các đối tượng, từ đó dự đoán khoảng cách một cách chính xác.

Mô hình GLPDepth + LSTM tận dụng cả thông tin không gian và thời gian để dự đoán khoảng cách từ camera đến các đối tượng. Sự kết hợp giữa GLPDepth và LSTM cung cấp một cách tiếp cận toàn diện và mạnh mẽ để ước tính khoảng cách trong môi trường xe tự lái.

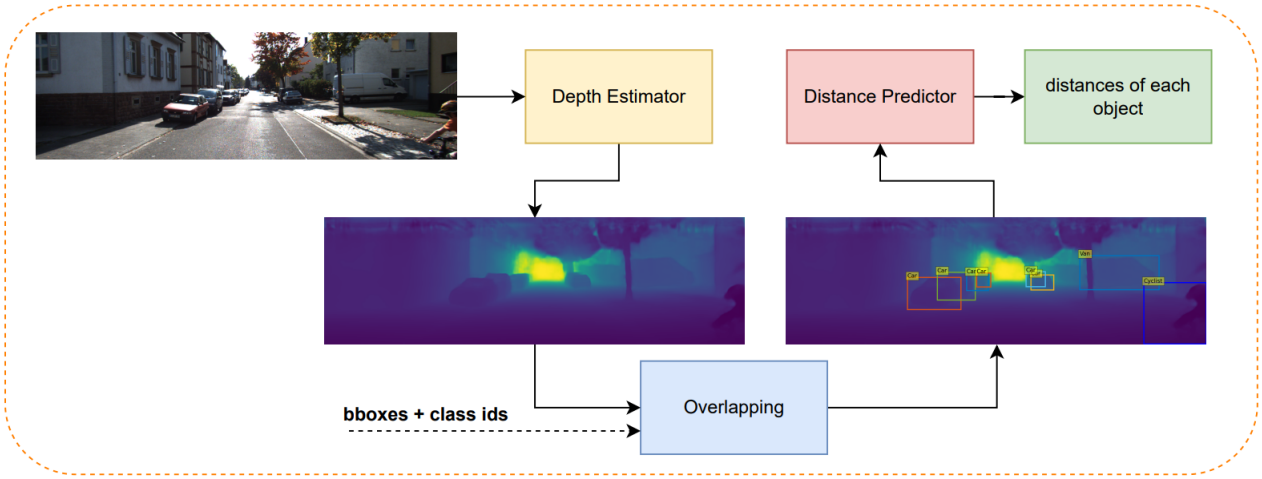

Figure 5: Tổng quan phương pháp ước tính khoảng cách.

3 Dataset

3.1 Thông tin bộ dữ liệu sử dụng

- Dataset: KITTI
- Số lượng: 7481 training images and 7518 test images
- Số lớp: 8 (Car, Pedestrian, Van, Cyclist, Truck, Misc, Tram, Person_sitting)

Figure 6: Ví dụ về một hình ảnh có trong bộ dữ liệu

3.2 Chuẩn bị dữ liệu cho ước tính khoảng cách

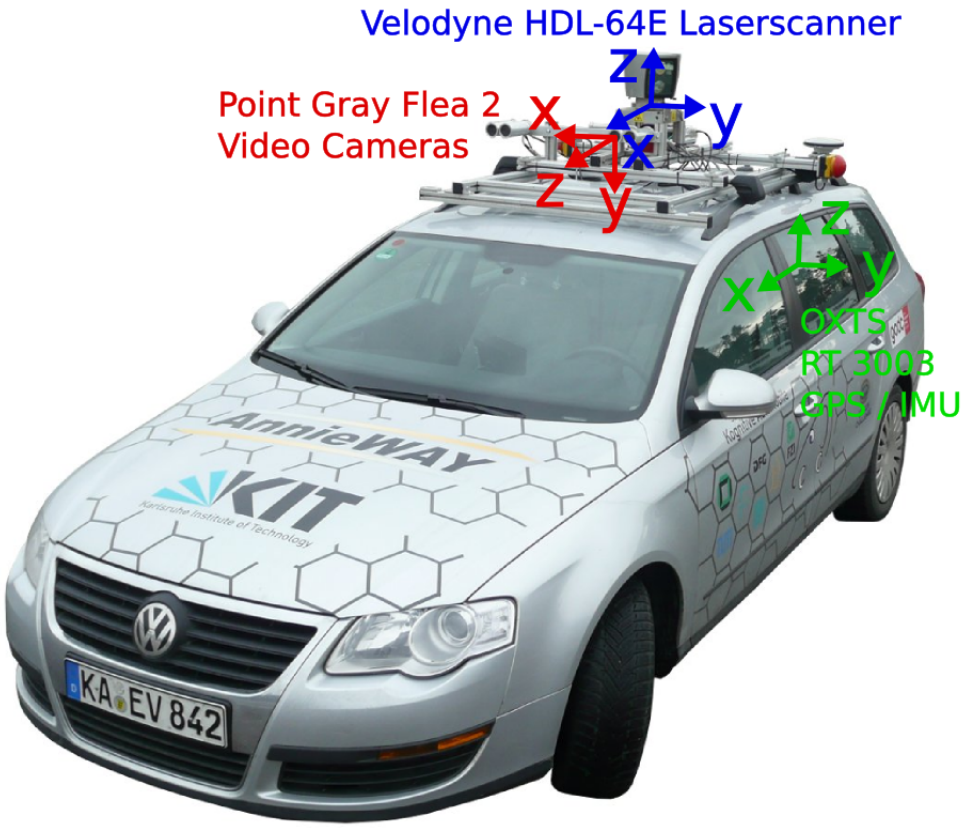

Figure 7: Thông tin setup sensor của bộ dữ liệu KITTI

Để chuẩn bị dữ liệu cho việc đào tạo mô hình dự đoán khoảng cách, nhóm sử dụng tập dữ liệu đã được gán nhãn, bao gồm các tọa độ (x_loc, y_loc, z_loc) của đối tượng đối với camera. Hệ tọa độ này được biểu diễn trong hình ở mức độ màu đỏ.

Hệ trục tọa độ gắn tại camera với $(x_{\text{cam}}, y_{\text{cam}}, z_{\text{cam}}) = (0,0,0)$. Sử dụng công thức sau tạo ra giá trị khoảng cách từ camera (gốc tọa độ) đến đối tượng dựa vào giá trị tọa độ (x,y,z) của

đối tượng:

$$\text{distance}_{\text{cam}->\text{obj}} = \sqrt{(x_{\text{obj}} - x_{\text{cam}})^2 + (y_{\text{obj}} - y_{\text{cam}})^2 + (z_{\text{obj}} - z_{\text{cam}})^2} \quad (1)$$

Phân phối dữ liệu khoảng cách

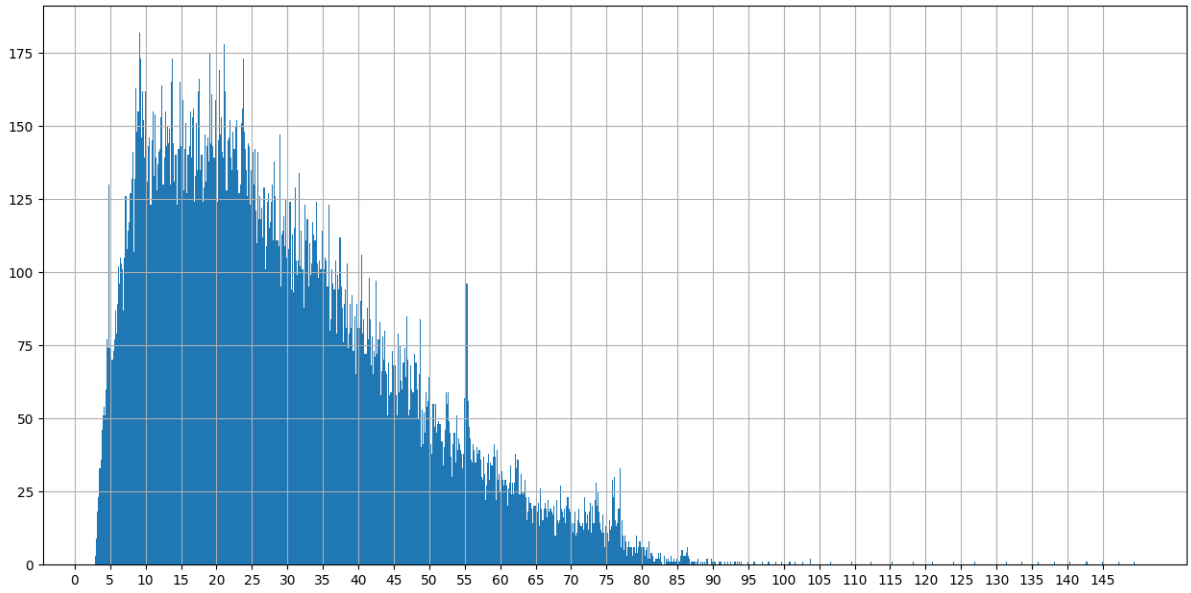

Figure 8: Phân phối dữ liệu khoảng cách

Nhìn vào biểu đồ phân phối khoảng cách sau khi thu thập dữ liệu ta thấy:

- Giá trị khoảng cách lớn nhất: 147(m)
- Giá trị khoảng cách bé nhất: 3(m)
- Dữ liệu có giá trị phân bố trong khoảng 7->30 (m), sau đó giảm dần

4 Quá trình đào tạo và kết quả thực nghiệm

4.1 Huấn luyện mô hình YOLOv5

Thông tin cài đặt huấn luyện mô hình YOLOv5

- Dữ liệu: 7481 ảnh
- Phân chia: train: 0.8, test: 0.2
- Pretrained: yolov5x.pt
- epochs: 100

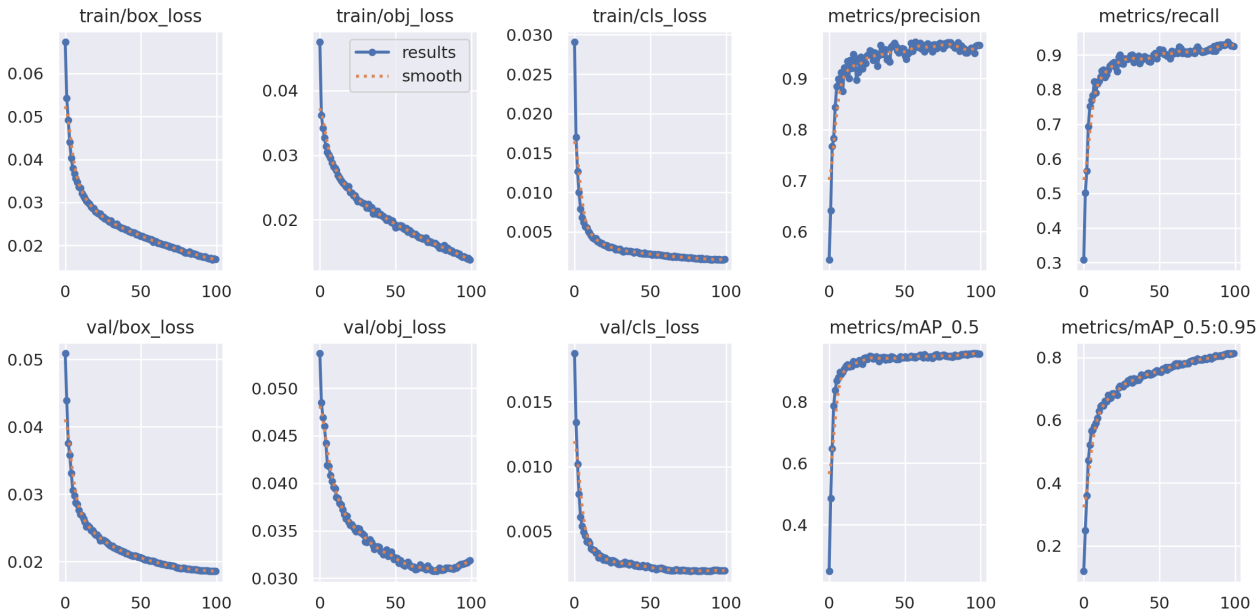

Figure 9: Kết quả huấn luyện mô hình YOLOv5

Class	Images	Instances	P	R	mAP50	z error far/m
all	749	4079	0.965	0.927	0.957	0.813
Car	749	2916	0.974	0.953	0.983	0.89
Pedestrian	749	446	0.94	0.886	0.927	0.608
Van	749	284	0.966	0.954	0.968	0.867
Cyclist	749	163	0.936	0.939	0.96	0.76
Truck	749	95	0.983	0.958	0.986	0.892
Misc	749	99	0.967	0.939	0.982	0.87
Tram	749	46	0.993	1	0.995	0.894
Person_sitting	749	30	0.959	0.784	0.853	0.721

Table 1: Kết quả đánh giá mô hình trên tập test

Kết luận: Ta thấy rằng kết quả huấn luyện mô hình YOLOv8 trên tập dữ liệu KITTI tốt với mAP@50 = 95.7%

4.2 Huấn luyện mô hình ước tính khoảng cách

4.2.1 Trích xuất đặc trưng

Khi ánh xạ tọa độ và class IDs vào bản đồ độ sâu (depth map), trích xuất các đặc trưng cần thiết để dự đoán khoảng cách, dưới đây là 9 đặc trưng được sử dụng:

Danh mục	Biến	Miêu tả
Input	x_min	Minimum x coordinate of a bounding box
	y_min	Minimum y coordinate of a bounding box
	x_max	Maximum x coordinate of a bounding box
	y_max	Maximum y coordinate of a bounding box
	width	Width of a bounding box
	height	Height of a bounding box
	depth_mean	Mean depth of an object
	depth mean_trim	20% trimmed mean depth of an object
	depth_max	Maximum depth of an object
Output	d	Distance of an object

Table 2: Các đặc trưng được sử dụng để huấn luyện mô hình

4.2.2 Cài đặt tham số cho mô hình LSTM

LSTM	Input_dim	10
	Hidden_dim(LSTM)	612
	Layer_dim(LSTM)	3
	Hidden_dim(Linear)	612, 306, 154, 76
	Output_dim(Linear)	1
	Bidirectional	False
	Optimizer	Adam
	Acitivation function	ReLU
	Max epoch	500
	Batch size	64

Table 3: Các siêu tham số và giá trị tương ứng cho mô hình LSTM

4.2.3 Loss function

Độ lỗi trung bình tuyệt đối (MAE), hay Mean Absolute Error, là một phép đo lường sự chênh lệch trung bình giữa giá trị dự đoán và giá trị thực tế. Đối với mỗi điểm dữ liệu, MAE tính giá trị tuyệt đối của hiệu giữa dự đoán và thực tế, sau đó tính trung bình của tất cả các giá trị này. MAE là một phương pháp phổ biến để đánh giá hiệu suất của mô hình dự đoán và đo lường mức độ chính xác trung bình của các dự đoán so với giá trị thực tế.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y - \hat{y}| \tag{2}$$

4.2.4 Thông tin cài đặt và kết quả đào tạo mô hình LSTM

Thông tin cài đặt huấn luyện mô hình LSTM

- Dữ liệu: 40570 mẫu
- Phân chia: train:0.8, valid:0.1, test:0.1
- max epoch: 500, sử dụng EarlyStopping

Kết quả huấn luyện mô hình LSTM

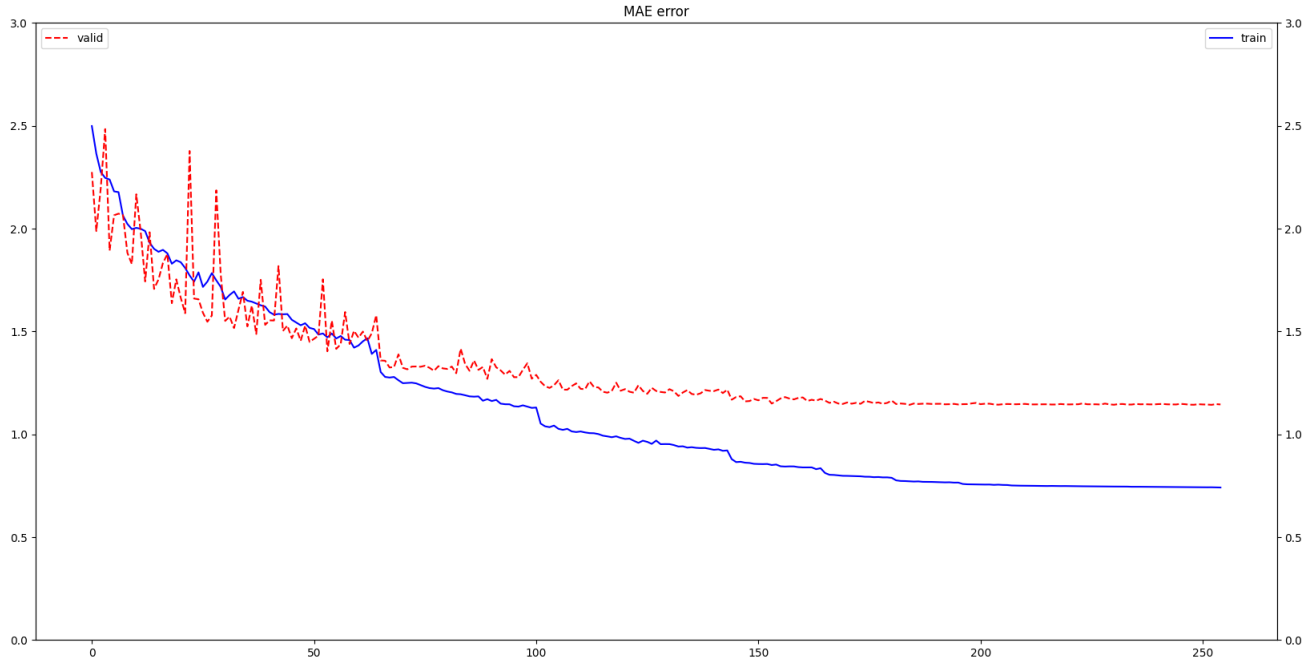

Figure 10: Độ lỗi MAE qua mỗi epoch

Kết quả đánh giá mô hình trên tập test

Model	MAE(m)								
	Car	Truck	Pedestrian	Van	Cyclist	Person_sitting	Tram	Misc	Overall
LSTM	1.16	1.48	0.78	2.07	1.09	1.45	1.46	1.34	1.20

Table 4: MAE cho các lớp đối tượng khác nhau trong tập test

Model	MAE(m)							
	0-9 m	10-19 m	20-29 m	30-39 m	40-49 m	50-59 m	60-69 m	70-79 m
LSTM	0.91	0.96	0.96	0.96	0.96	0.96	0.92	0.89

Table 5: MAE cho các khoảng khoảng cách khác nhau trong tập test

5 Tổng kết

Tổng kết quá trình tìm hiểu và thực nghiệm đối với bài toán theo dõi và ước tính khoảng cách đối tượng trong xe tự lái, nhóm có đưa ra một số nhận xét đối với các mô hình sử dụng

Đối với mô hình phát hiện đối tượng

- Độ chính xác cao trong viêc phát hiện đối tượng xe hơi, người đi bộ,..
- Có thể chạy real-time

Đối với mô hình theo dõi

- Theo dõi chính xác và nhất quán các đối tượng qua các frame
- Ít xảy ra hiện tượng đánh tráo đối tượng

Đối với mô hình ước tính khoảng cách

- Mô hình đào tạo có độ chính xác khá cao
- Tốc độ inference còn chậm do mô hình ước tính độ sâu GLPDepth có thời gian thực thi chậm
- Bộ dữ liệu đào tạo không cân bằng, làm cho mô hình dự đoán sai khi chạy thử

Qua quá trình tìm hiểu này, nhóm có đưa ra một số dự định cải tiến trong tương lai

- Tìm kiếm và thu thập thêm dữ liệu cho bài toán ước tính khoảng cách
- Cải thiện mô hình ước tính khoảng cách bằng cách tăng tốc độ xử lý so với phương pháp nhóm sử dụng hiện tại, hướng đến việc chạy real-time cho bài toán này.
- Có thể tích hợp thêm nhiều bài toán khác như phát hiện làn được, báo động va chạm,..

6 Tham khảo

- https://github.com/ultralytics/yolov5
- https://github.com/dyhBUPT/StrongSORT
- https://arxiv.org/pdf/2202.13514.pdf
- https://arxiv.org/pdf/1905.00953.pdf

- https://github.com/KaiyangZhou/deep-person-reid
- https://github.com/vinvino02/GLPDepth
- https://www.mdpi.com/2073-8994/14/12/2657