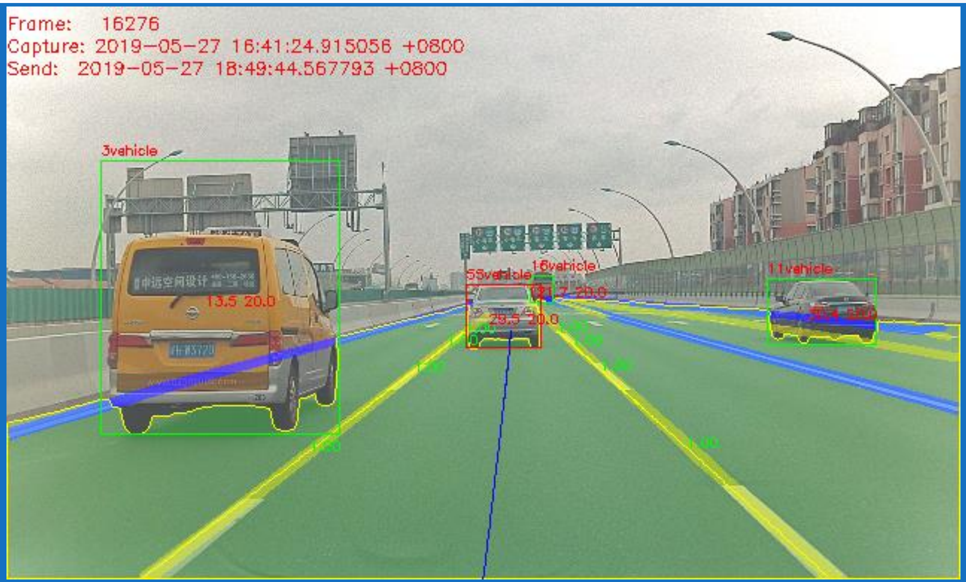
End-to-End Learning

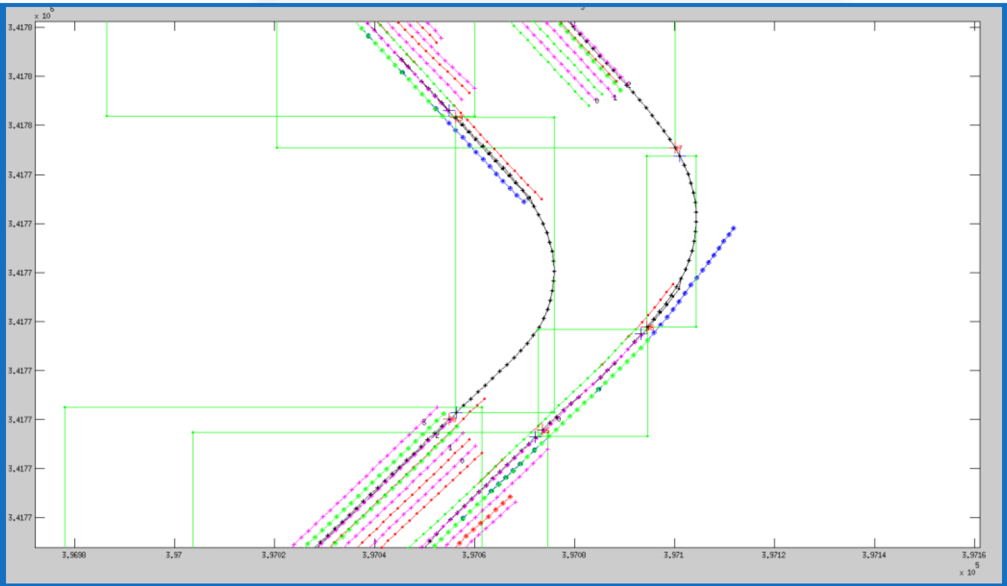
Yang Yuan
Machine Learning Department

Agenda

Background

Current Solution of Self-Driving





Lanes Detection

Obstacle Detection

Route Planning

High-precision Map

Limitation

Facilities	Computing Cost	Complicated Planning	Participation of Driver
• U-blox • RTK	• Large-scale data parallel computing	• Real-time route planning	• Dependency of reference • Need of taking over

Introduction

02 INTRODUCTION

End-to-End learning Demo

02 INTRODUCTION

02 INTRODUCTION

What Is End-to-End Learning

- A kind of deep learning process
- Joint training process rather than step by step

- Imitation of human behavior
- Human acceptable input

- Complete vision solution
- Removal of intermediate process

- Output of direct vehicle-control command

Pros

- Omit intermediate complex process
- Similar to human driving behavior
- Output can be directly used

From result

- Get rid of unnecessary sensing facilities
- Reduce cost for computing
- Reduce dependency on external references

From processing

Feasibility

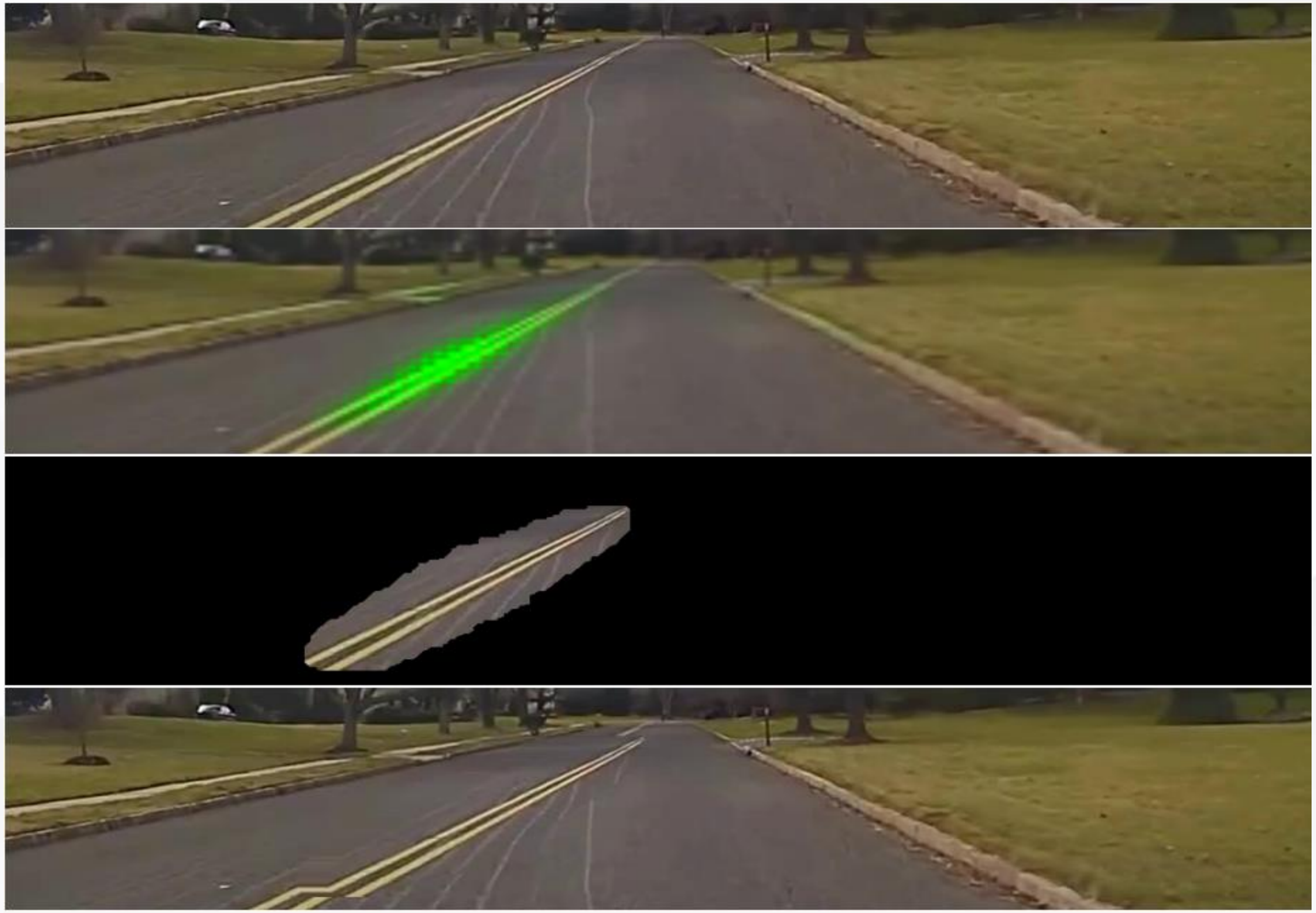


Figure. 1.1: Images used in experiments to show the effect of image-shifts on steer angle

Feasibility

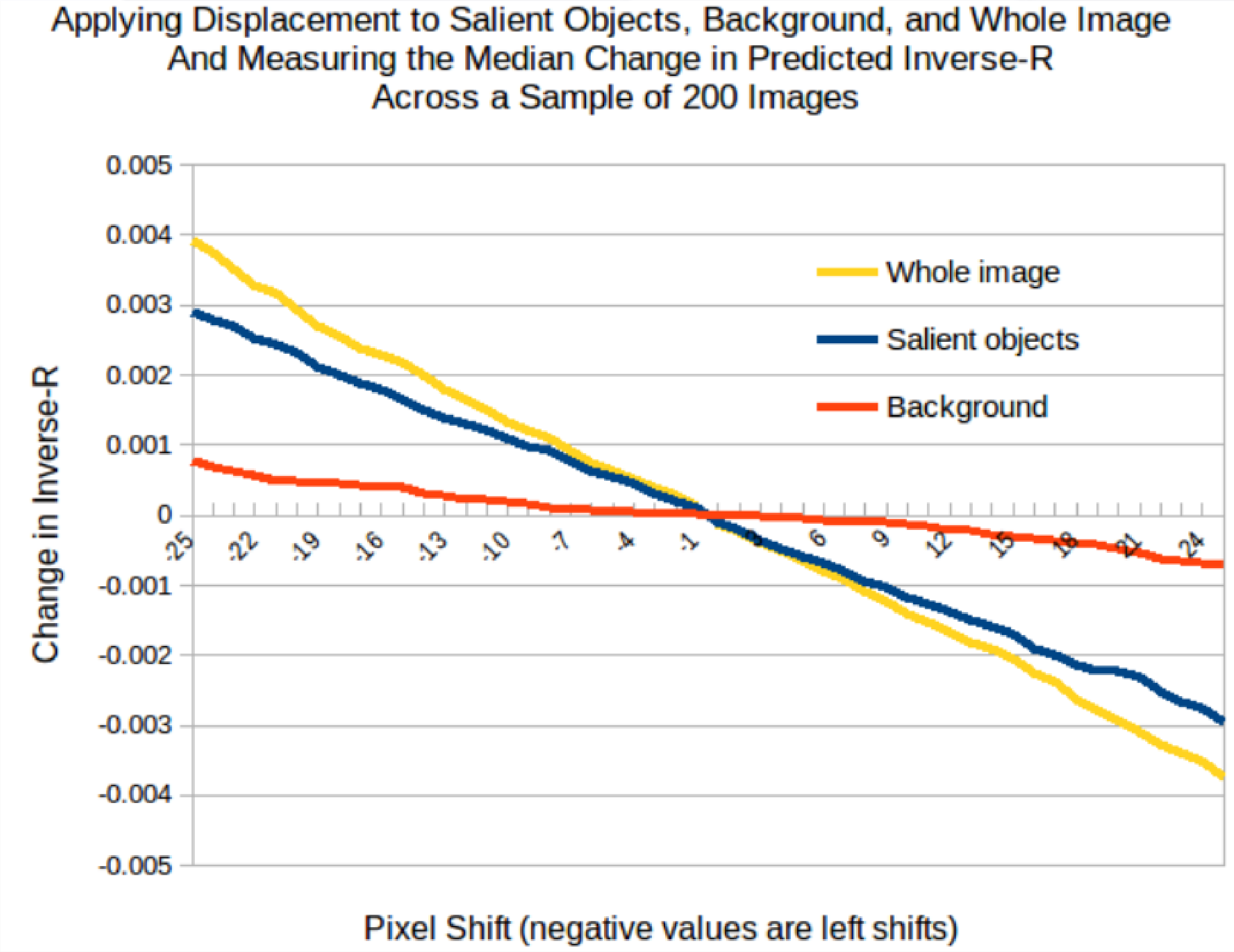
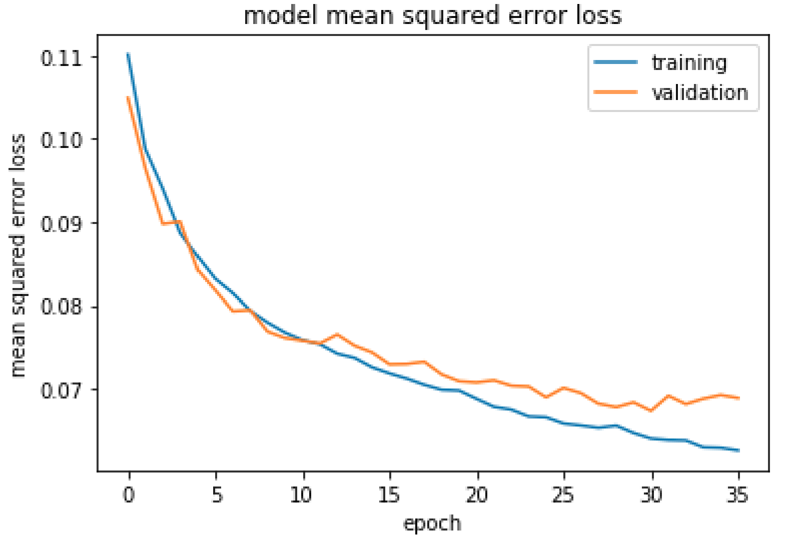


Figure. 1.2: Plots of PilotNet steering output as a function of pixel shift in the input image

02 INTRODUCTION

Feasibility

(i) Left (ii) Centre (iii) Right camera image



Ta=raining vs Validation loss

Core Technology

Network Architecture

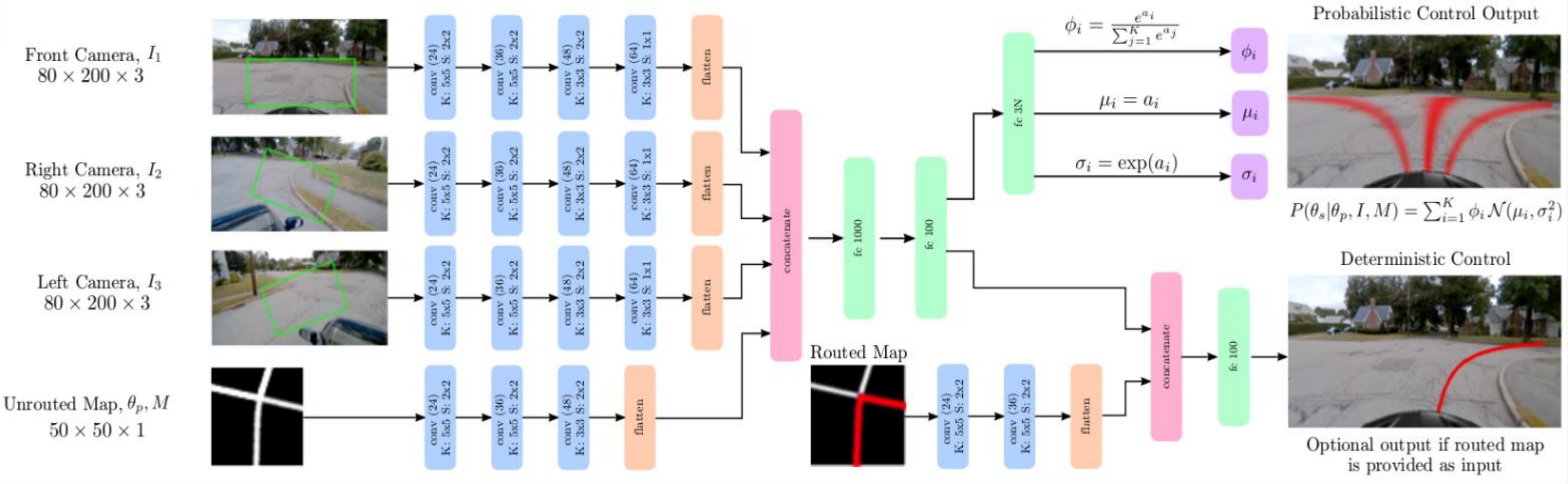


Fig. 2.1 Model architecture
Source: Amini A., et al. Variational End-to-End Navigation and Localization

Camera Model

Figure. 2.2: Images from camera input

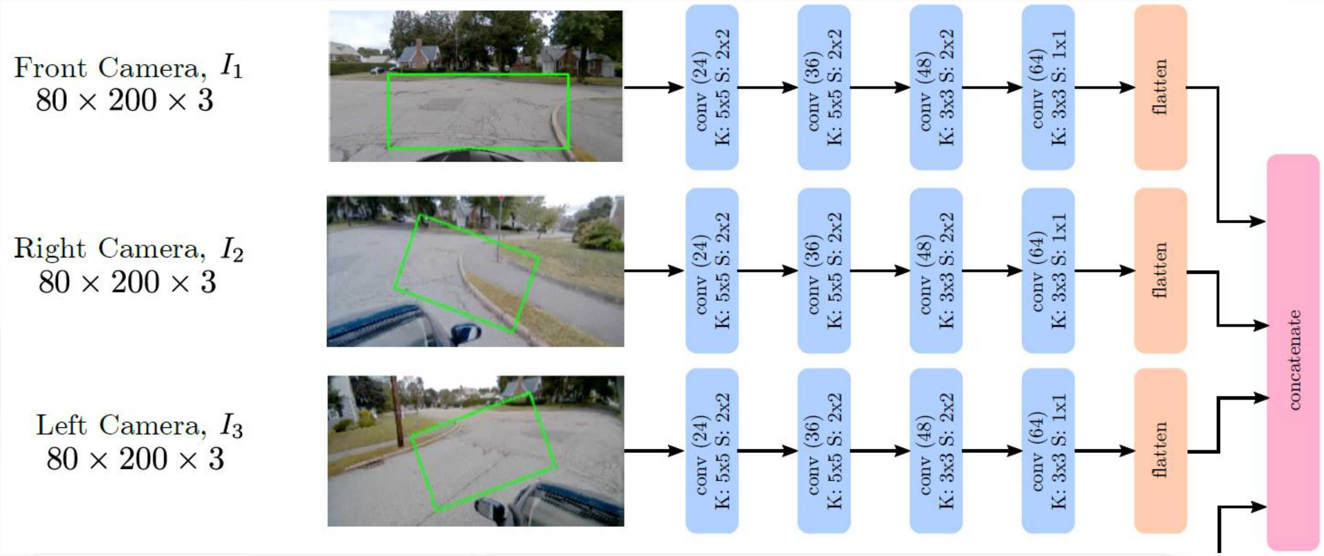


Figure. 2.3: Camera model

Map Model

- Merge heading information
- Guide direction from a preset route
- Implement localization

Figure. 3.4: Map input cropped by osmnx

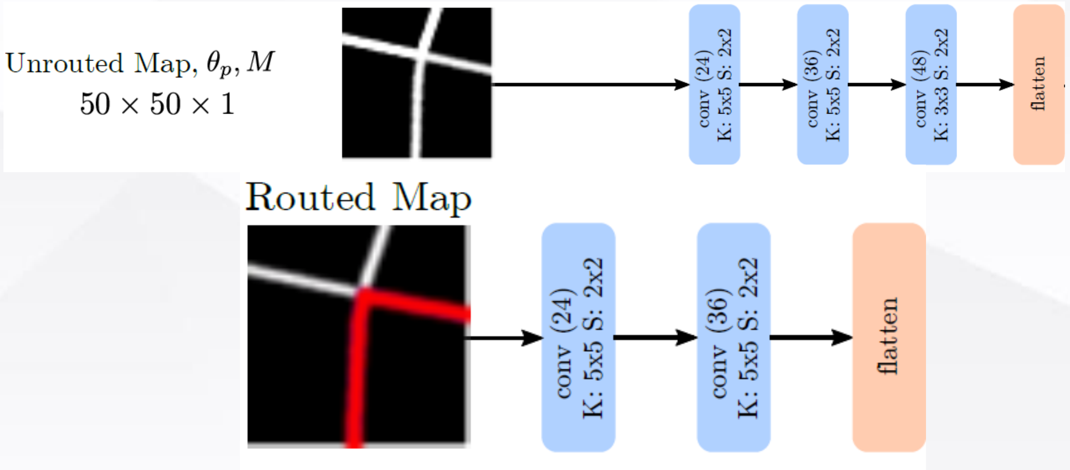


Figure. 3.5: Map model

Gaussian Mixture Model

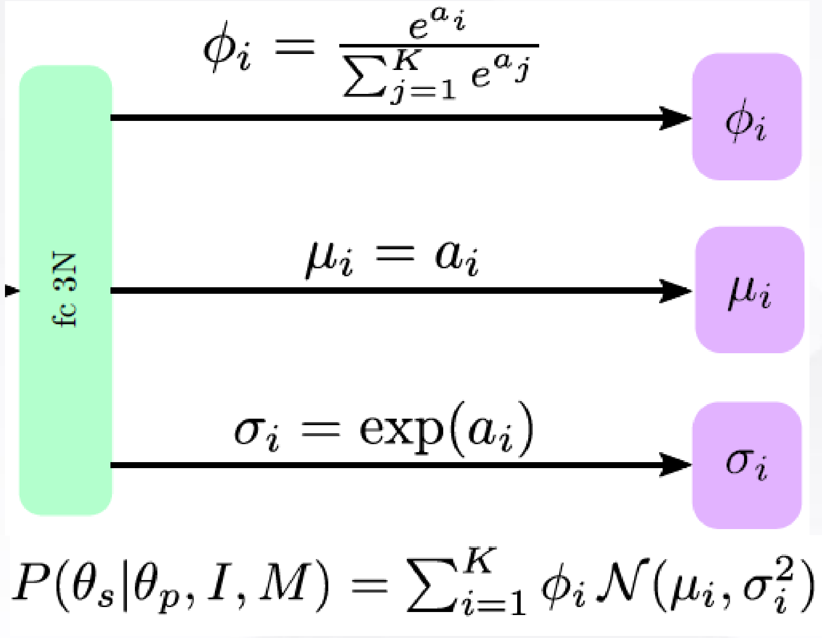


Figure. 3.6 GMM model

- **Assume N Gaussian clusters to estimate available roads**
- **Use fully connected layers to simulate GMM**
- **Fit three parameters to generate probabilistic control**

Loss Function

$$\left\{ \begin{array}{c} \mathcal{L}\Big(f_S(I, M, \theta_p), \theta_s\Big) + \|\phi\|_p + \\ \sum_i \psi_S(\sigma_i) + \Big(f_D(I, M, \theta_p)| - \theta_s\Big)^2 \end{array} \right\}$$

Log likelihood of PDF

L1 norm of weight

L2 norm of variance

MSE of curvature

Experiment

Input Data

Figure. 4.1: Origin map

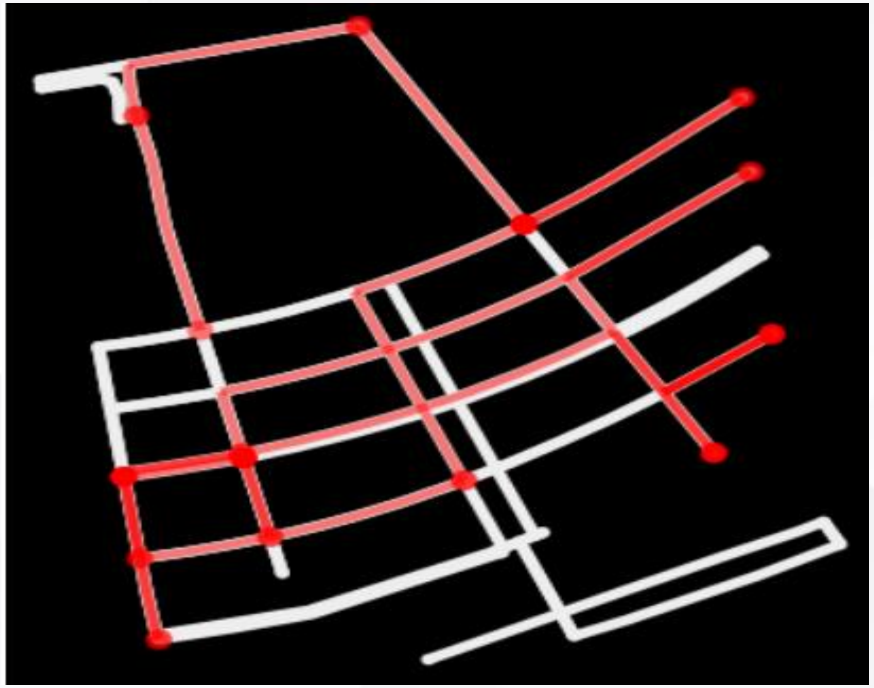


Figure. 4.2: Map projected by osmnx

Input Data

Figure size: 200 * 80 * 3

Figure. 2.2: Images from camera input

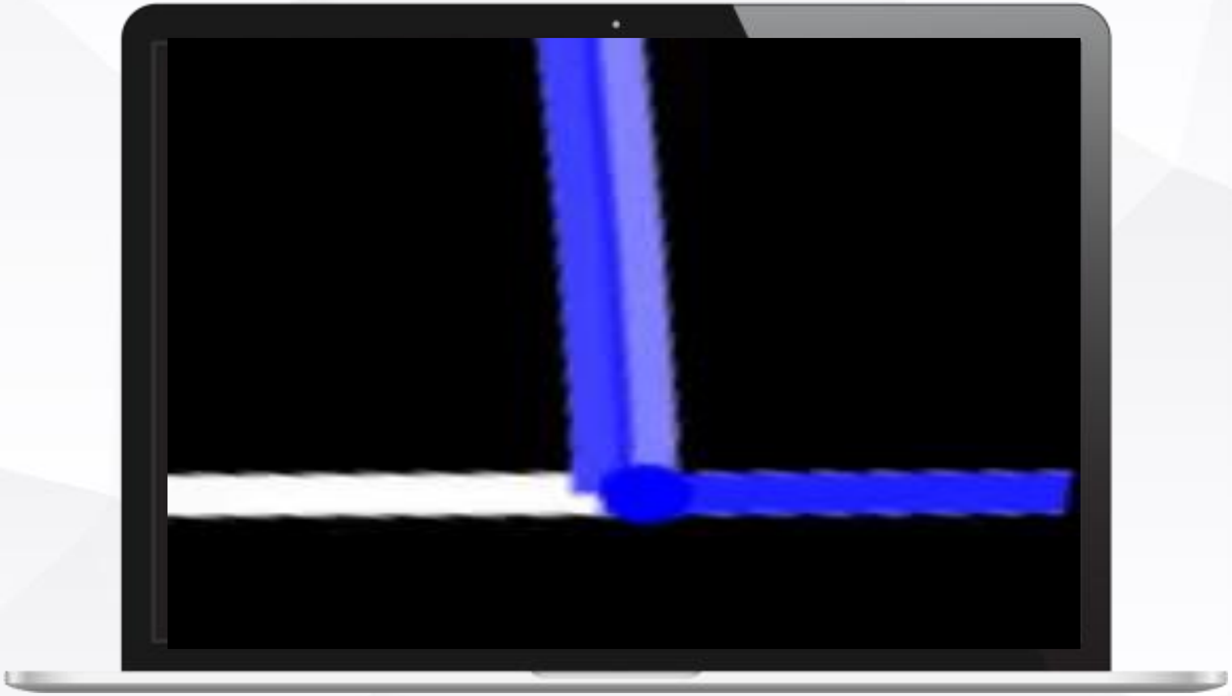
Figure size: 50 * 50 * 1

Figure size: 50 * 50 * 3

Figure. 4.4: Map input cropped by osmnx

Input Demo

Camera



Map

Result

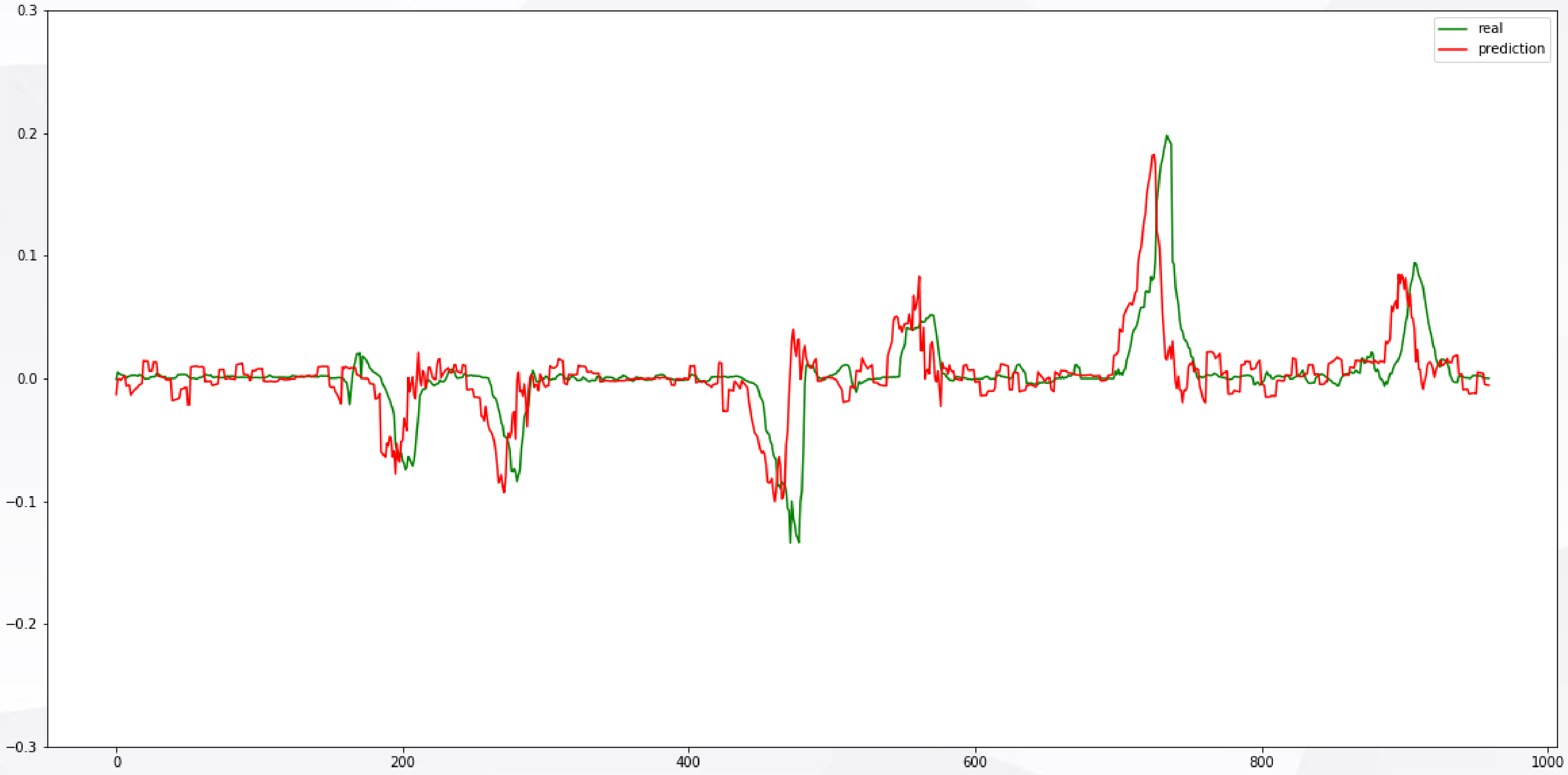


Figure. 4.5: Prediction on training set

R2 score: 0.86

Result

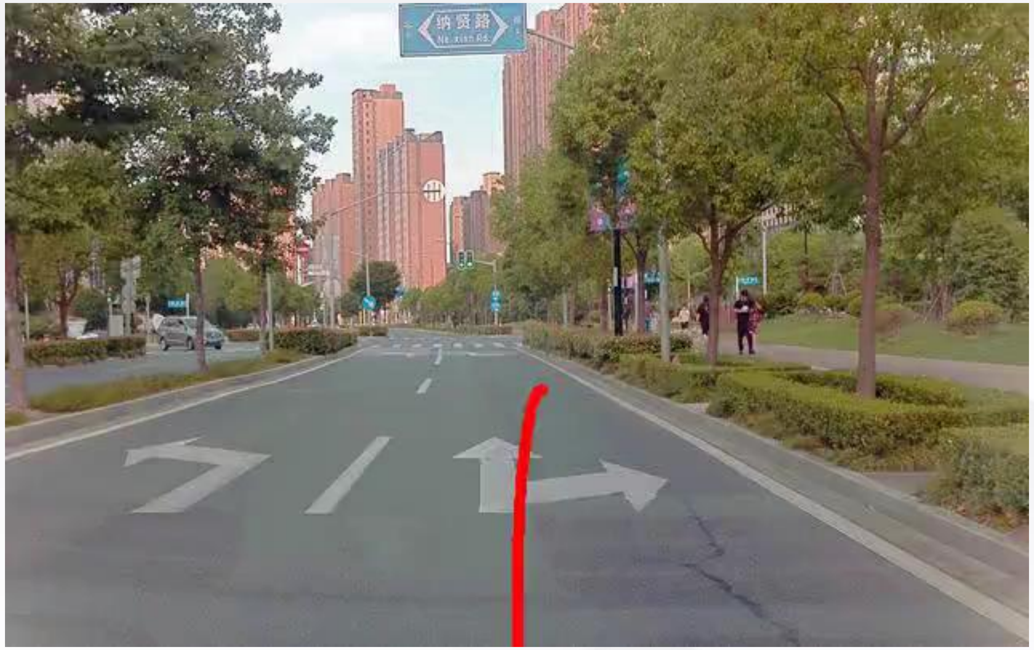


Figure. 4.6: Prediction on training set

Figure. 4.7: Prediction on testing set

Conclusion

Affirmative Result

- Get an accurate result
- Realize localization
- Preset route

Future for L4/L5

Feasibility

Accuracy

Simplification

Future

Verify the solution

- Reduce intermediate processing
- Get rid of high-precision facilities/maps

Limitation

- Lower performance on complicated scene
- Need of large training data
- Problem of speed

THANK YOU

Machine Learning Department