This is The Way: Vision-Based End-to-End Planning for Autonomous Driving

Anisha Palaparthi[1], Arpit Dwivedi[1], Purushottam Mani[1]

anishapv@stanford.edu, dwivedi7@stanford.edu, purush@stanford.edu

Stanford Computer Science

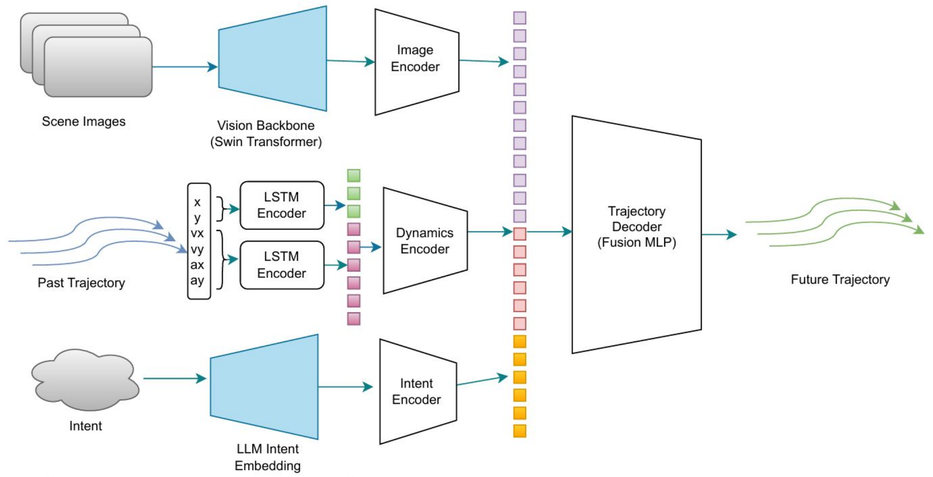
Introduction

- Autonomous driving through deep learning has attracted substantial interest in recent years
- End-to-end autonomous driving replaces complex, multi-module pipelines with a single neural network that directly predicts a vehicle's future trajectory. By fusing past motion states, current onboard camera images, and a high-level driving intent, our approach:
  - **Simplifies integration**.
  - **Improves generalization** to rare "long-tail" scenarios.
  - **Ensures platform agnosticism**, adapting readily across different vehicle platforms.

Data & Features

- **Inputs**: Each sample consists of 3 RGB views (front, front-left, front-right), 16-step past vehicle dynamics over 4s @ 4 Hz, driving intent: "go left", "go right", "go straight", or "unknown"
- **Prediction**: 20 future (x, y) positions over 5s @ 4 Hz; preprocessing of data includes CLIP normalization, random crop, and color jitter: Resize → crop → normalize (CLIP stats)

Methodology



~49.5M parameters, trained on Smooth L1 loss for robustness in noisy, low-data regimes.

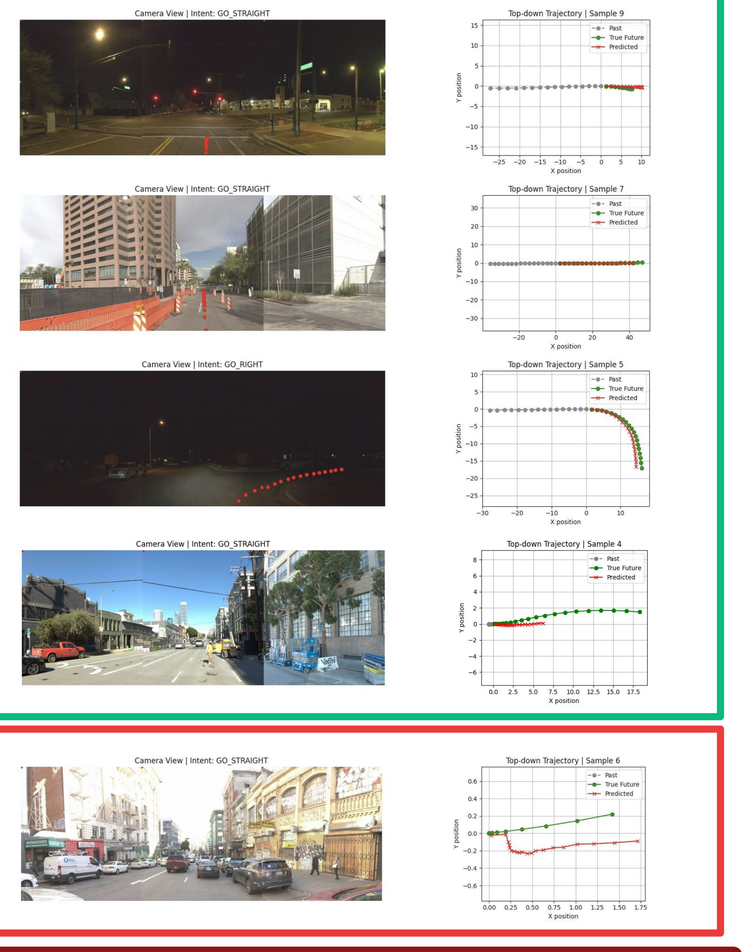
(1) **Vision Backbone -** A lightweight Swin Transformer pre-trained on ImageNet and fine-tuned on dataset, encodes the 3 images into a 256-dimensional scene embedding.

(2) **Dynamics Encoder**: Two LSTMs (for position and kinematics) encode motion history, followed by an MLP to produce a 64-dimensional dynamics embedding.

(3) **Intent Encoder**: Natural language prompts (e.g., "Go left while avoiding obstacles") are embedded using MiniLM, passed through an MLP to yield a 16-dim intent representation

(4) **Fusion Module**: The three embeddings are concatenated and passed through a 2-layer MLP to regress a 20×2 trajectory (x, y coordinates at 4Hz for 5 seconds).

Results



Experiments

MSE Loss

$$L_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N}\Big[\,(t_{i,x}-\hat{t}_{i,x})^2 + (t_{i,y}-\hat{t}_{i,y})^2\,\Big]$$

Smooth L1 (Huber) Loss

$$L_{\text{SmoothL1}} = \frac{1}{N}\sum_{i=1}^{N}\Big[\text{smooth}_{\text{L1}}(t_{i,x}-\hat{t}_{i,x}) + \text{smooth}_{\text{L1}}(t_{i,y}-\hat{t}_{i,y})\Big]$$

$$\text{smooth}_{\text{L1}}(r) = \begin{cases} 0.5r^2, & \text{if } |r|<1 \\ |r|-0.5, & \text{otherwise} \end{cases}$$

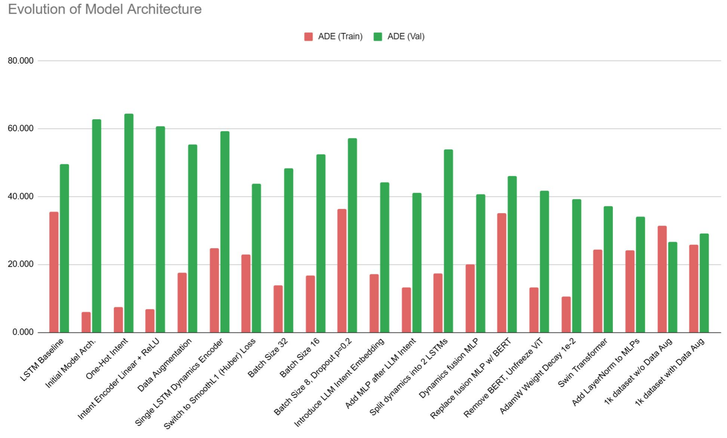
Average Displacement Error (ADE) Metric

$$\text{ADE}(t,\hat{t}) = \frac{1}{N}\sum_{i=1}^{N}\Big[(t_{i,x}-\hat{t}_{i,x})^2 + (t_{i,y}-\hat{t}_{i,y})^2\Big]$$

Train/Val/Test: 1k train samples (250 for model development exps), 96 val, 263 test samples

Summary of Experiments:

	Train Loss	Val Loss	ADE-T	ADE-V
Baseline	1.235	4.198	35.600	49.521
Our Model	0.606	0.663	25.937	29.237
Final ADE of Model on Test Set: **26.96**				



Conclusion & Future Work

- **Conclusion:** Our model architecture successfully outperforms baseline no-vision LSTM, generalizes well to plan trajectories in new scenarios, and shows potential for further improvement with more data/compute
- **Unified Vision–Language Encoding:** Replace frozen Swin and MiniLM modules with a joint VLM (e.g., Dolphins) to learn shared embeddings for images and intent, improving cross-modal alignment.
- **Diffusion-Based Trajectory Generation:** Try adopting conditional diffusion models on fused visual, motion, and intent features to capture uncertainty and produce diverse, multimodal future paths.

References: [1] M. Bojarski et al., "End to End Learning for Self-Driving Cars,", 2016 [2] Waymo LLC, "Waymo Open Dataset: Vision-Based End-to-End Driving Challenge (2025)," [3] Loshchilov and F. Hutter. Decoupled weight decay regularization, 2017