CS M213A / ECE M202A (Fall 2025)

Context-Aware Hybrid Object Detection for Autonomous Vehicle Perception

Hee Jean Kwon

Motivation and Objectives

- Autonomous Vehicles need to detect objects quickly and accurately
- Running powerful vision models on the vehicle reduces latency but limited by hardware -> using the cloud gives better accuracy but causes delay
- A runtime optimizer that can decide when to use local vs cloud processing
  - Faster and safer decision for autonomous vehicles
  - More efficient use of limited onboard resources
- Goals
  - A context-aware decision system that choose between local and cloud model in real time
  - Run demo/simulation using Jetson and CARLA

Technical Approach and Novelty

- Current Practices
  - Offloading uses a single factor decisions rules
    - either focusing on networks or compute load
  - Evaluations are offline and system metrics
    - Does not incorporate closed-loop driving tests

Technical Approach and Novelty

- Use multi-modal context (scene, vehicle, and system) to predict best model
- Investigate different decision architectures to identify optimal model selection
- Run End to End Scenario Simulation for evaluation

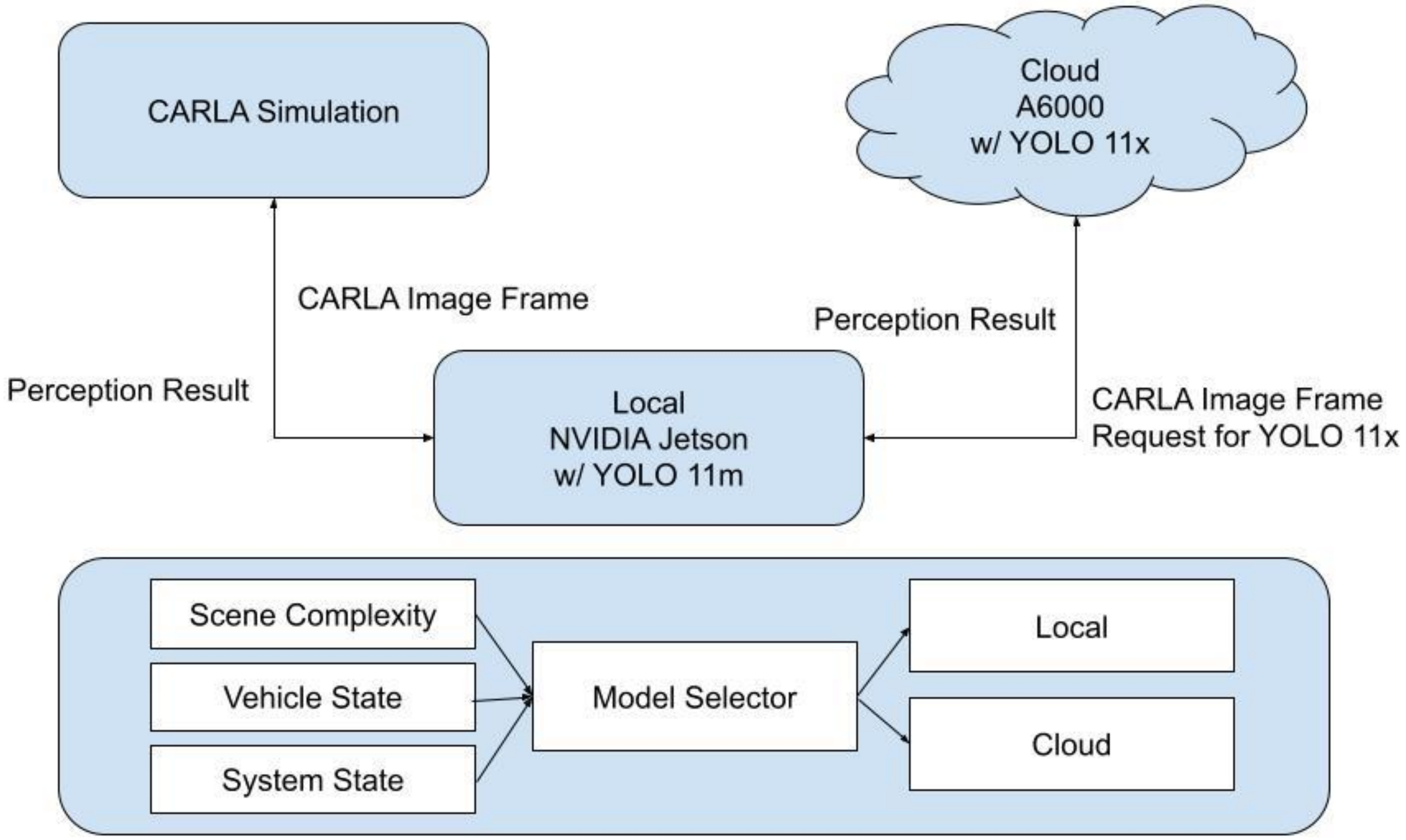

Figure 1 System overview

Methods

- Data Set
  - Waymo Perception Data set
- Platform
  - NVIDIA Jetson for local processing
  - A6000 for cloud processing
  - CARLA for simulation
- Use MLP/VLM to design a model selector
  - Scene complexity: Number of vehicles and pedestrians, brightness
  - Vehicle state: ego vehicle speed
  - System state: Network reachability and Cloud availability

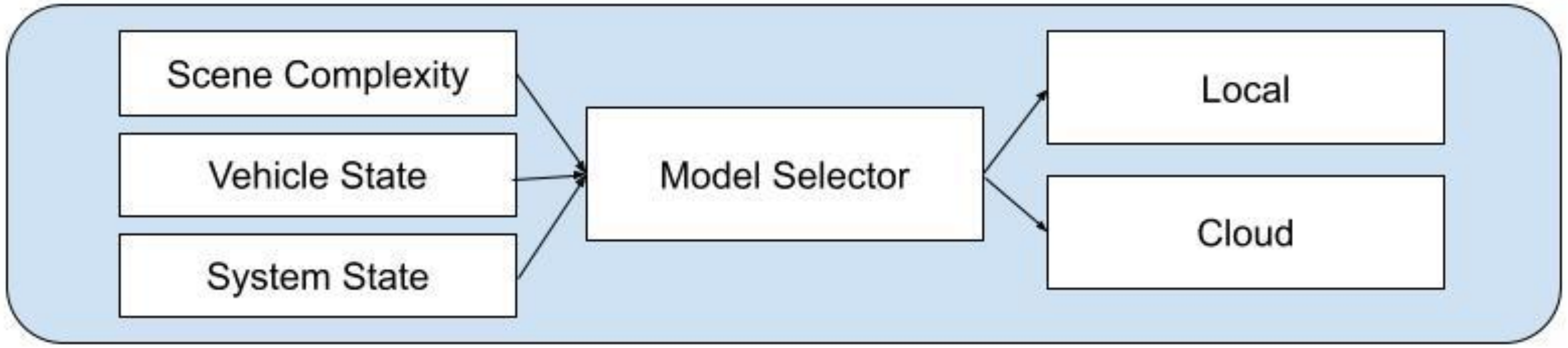

Figure 2 Model Selector Overview

Evaluations

- End to End Scenario Evaluation
  - Defined 5 different scenarios varying traffic density, number of pedestrians, and brightness
- Evaluation Metrics
  - System Efficiency: Inference Latency, End to End Latency, CPU/GPU Utilization
  - Driving Safety: Collisions Rate, Reaction Time
  - Comparison : always local vs always cloud

Current Status and Next Steps - MLP

- Inputs
  - # of vehicles, # of pedestrians, brightness, and ego vehicle speed
- Outputs
  - 0 = simple scene, 1 = complex scene
- Misclassified scenes are on boundary
- Feature Importance
- 28% # of vehicles, 20% ego vehicle speed, 15% # of pedestrians

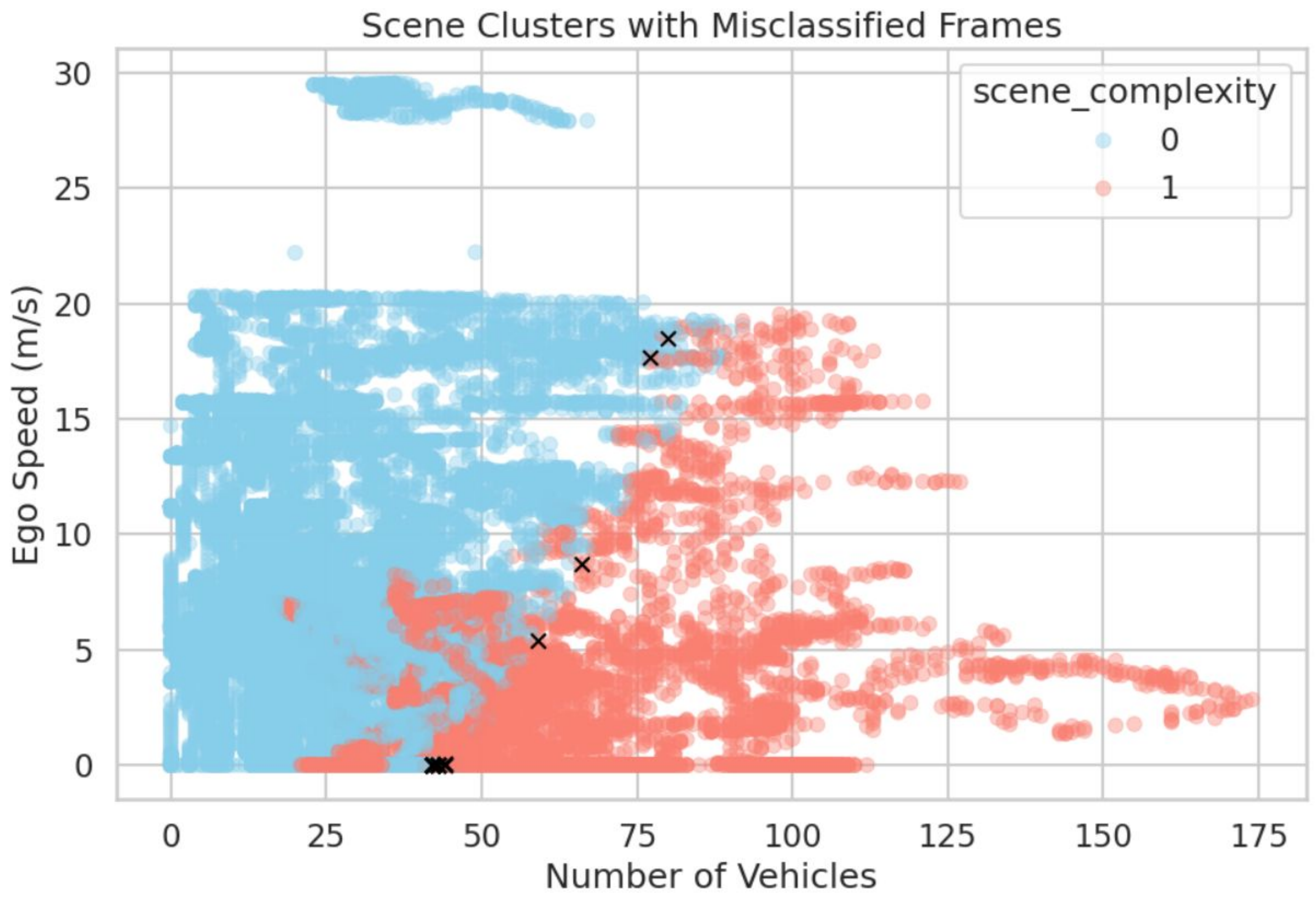

Figure 3 Scene Clusters with Misclassified Frames on MLP

Current Status and Next Steps - VLM

- Qwen2-VL with 2B parameters
- Inputs: JPEG image of one frame from Waymo dataset
- Output: Local or Cloud
- Prompt VLM to check the amount of vehicles, pedestrians, and brightness of the given image and make a decision
- Brightness has the strongest influence unlike MLP
- Its reasoning and the scenario do not match

Reasoning: The scene is complex with multiple vehicles and pedestrians, indicating a busy street, which requires processing tasks in the cloud for real-time safety and accuracy.

Figure 4 Example of an image frame with its VLM reasoning

Current Status and Next Steps

- Next Steps
    - Integrate system states input to the model selector
    - Integrate the model selector to the system
    - Run experiments across different scenarios and analyze results