ODAF: Out-of-Distribution Detection and Fallback for Autonomous Vehicles

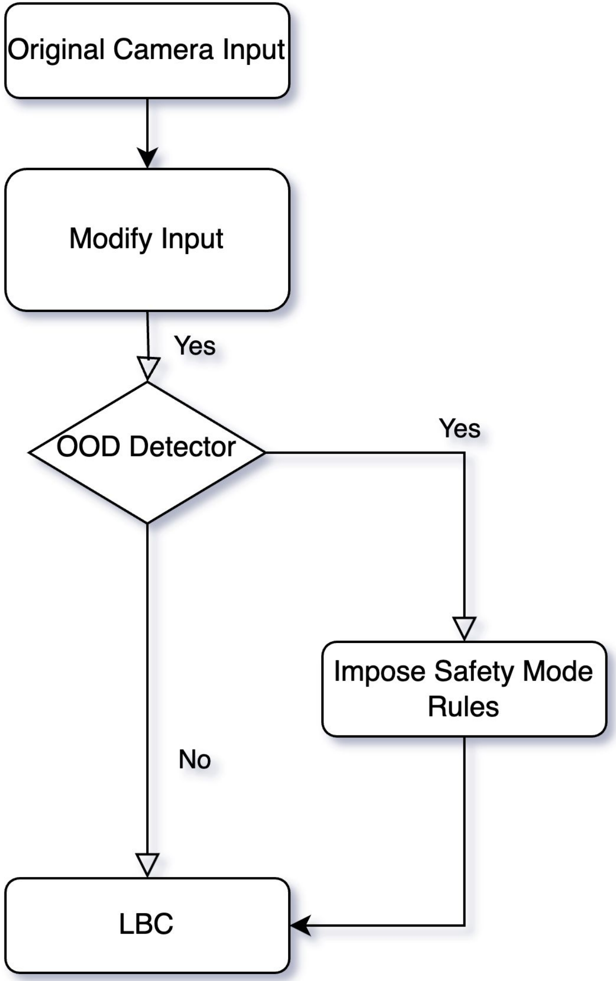
Jack, Barney, Weihang, Heramb

Motivation

- What is OOD in general?
  - Testing data point does not belong to the training dataset
- What is OOD for AVs:
  - Unpredictable Actions
  - Environmental Challenges
- What is OOD in our scope
  - Varying weather conditions impacting visual sensors
    - Industry pushing to visual-based ADS
- Learning-based approach does not work well
  - Our approach: experiment with other methods for OOD detection

Pipeline

1. Modify camera input with given parameters
2. Pass through OOD detection mechanism
3. Feed the modified camera input to LBC agent
    a. State-of-the-Art vision-based ADS
4. If OOD, enable fallback mode
    a. When applying controls, limit throttling and encourage braking



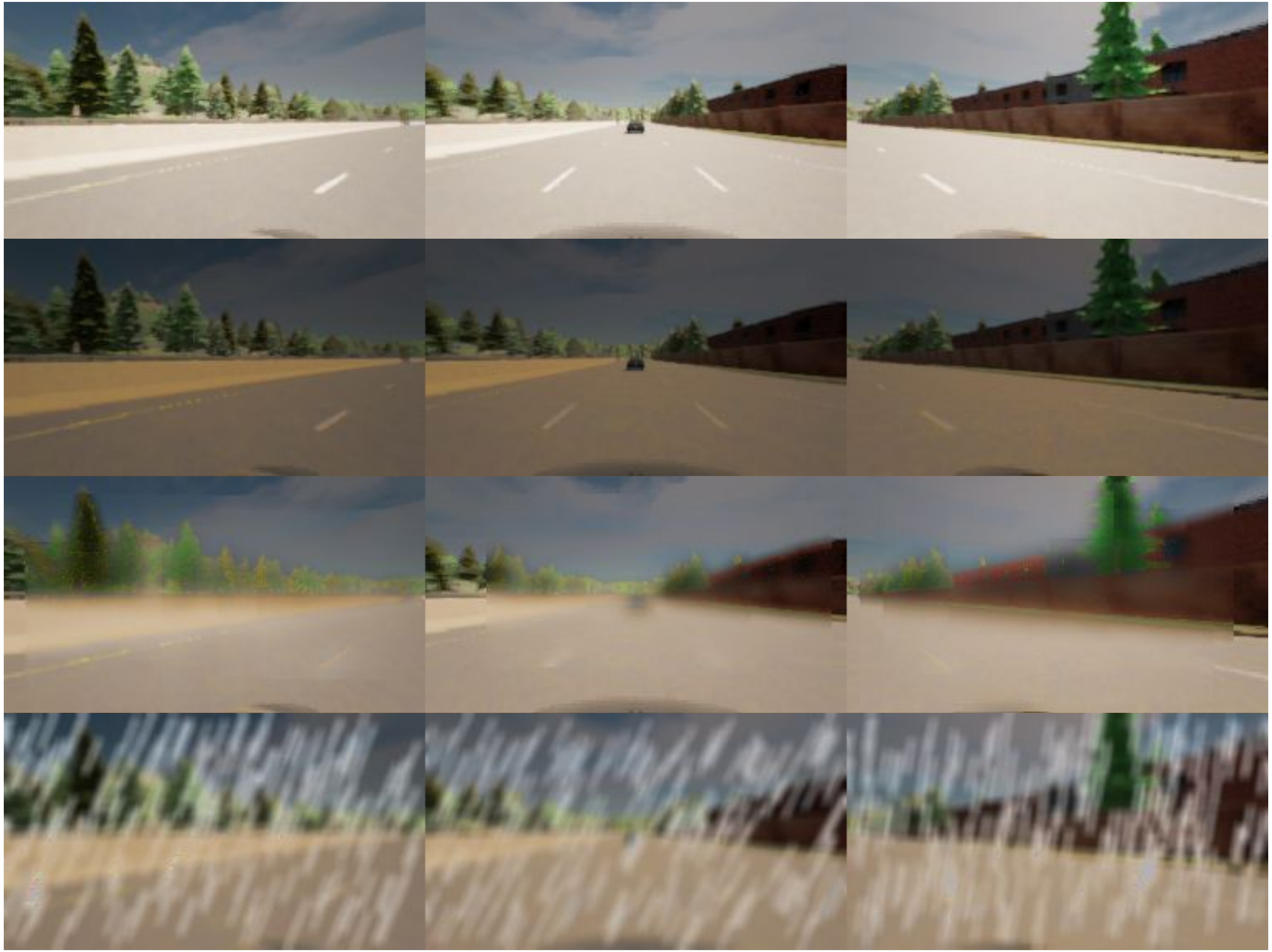
Comparing with Native Weather Parameters

Camera input in native “Heavy Rain” weather in CARLA

Our modification

Camera Input Modification

- Rain
  - Parameters: **rain intensity**
    - Get randomised image drops from a pool of hard-coded drops
    - Add drops to image such that: number of drops = intensity, alpha for each randomized
- Shade
  - Parameters: **shadow intensity**
    - Use shadow_intensity as a translucent image with alpha = intensity
    - Combine existing and translucent image
- Haze
  - Parameters: **haze intensity**
    - Draw a random number of random sized square blocks with white bg and alpa = intensity
    - From centroid of each square, convolute the haze square with image using a edge smoothing algorithm



OOD Detection via Mahalanobis (Maha) Distance

$$d_M(\vec{x}, Q) = \sqrt{(\vec{x} - \vec{\mu})^\mathsf{T} S^{-1} (\vec{x} - \vec{\mu})}.$$

- Intuition/summary
- What is the input data?
- Implementation: using covariance matrix, mean vector
- Challenges: too many data points -> extremely slow matrix construction (time complexity O(n^2*m))
- Solution: use PCA model to reduce dimension. Retained 95% of variance
- OOD only if all three images are OOD

OOD Detection via Autoencoders

- **Intuition:** Learning neural networks that capture essential features and discard noise.
- **Structure** (12 layers in total)**:**
  - **Encoder:** 3 linear layers & 3 activation layers.
    - Input: 1D vector of size 147456
    - Output: 1024 features -> 512 features -> 256 features
  - **Decoder:** 3 linear layers & 3 activation layers.
    - Input: 256 features
    - Output: 512 features -> 1024 features -> 147456 features
- **Model Use:** Trained to contrast with newly fed images for out-of-distribution OOD detection, with a predetermined threshold.
- **Detection:** OOD only if all three images are OOD.

OOD Detection Testing Strategy

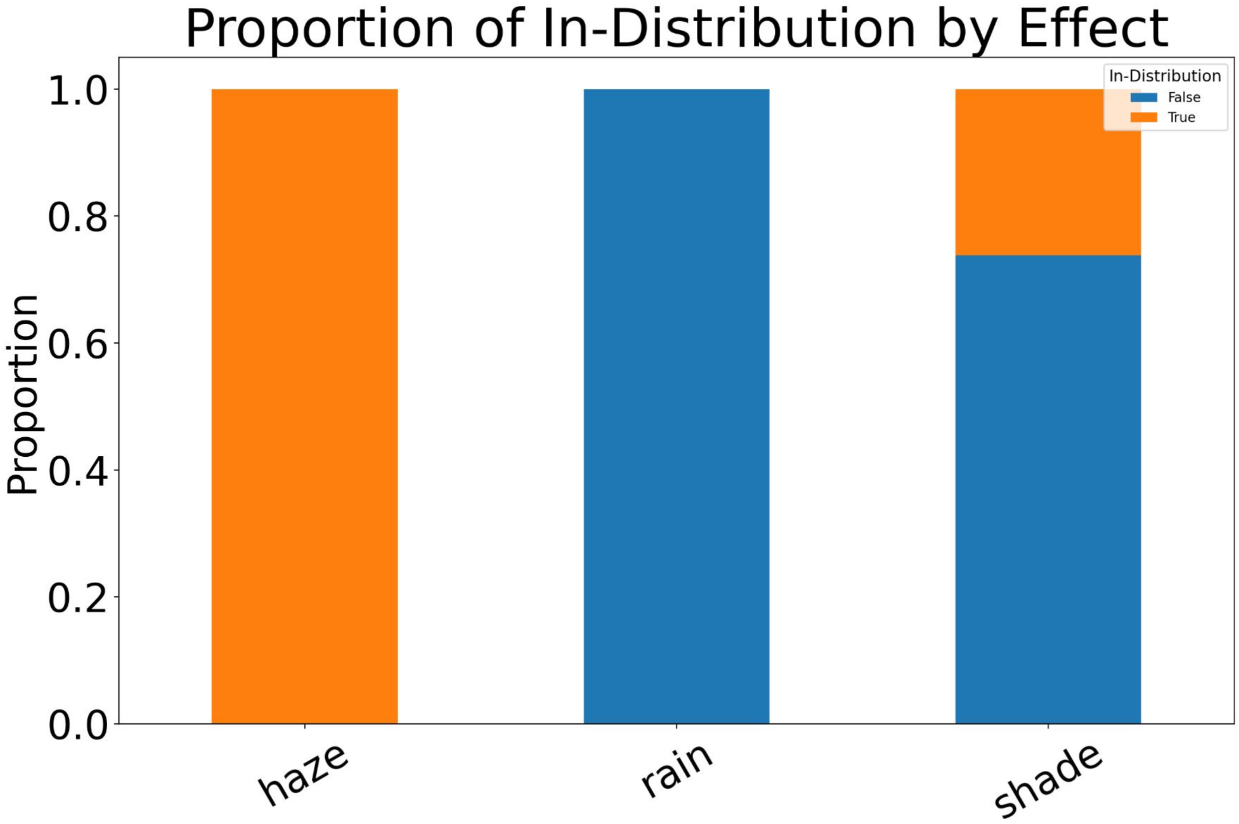
Does OOD detection work on unmodified images? (False Negative)

- Tested on all unmodified images
- **All images are in distribution!**

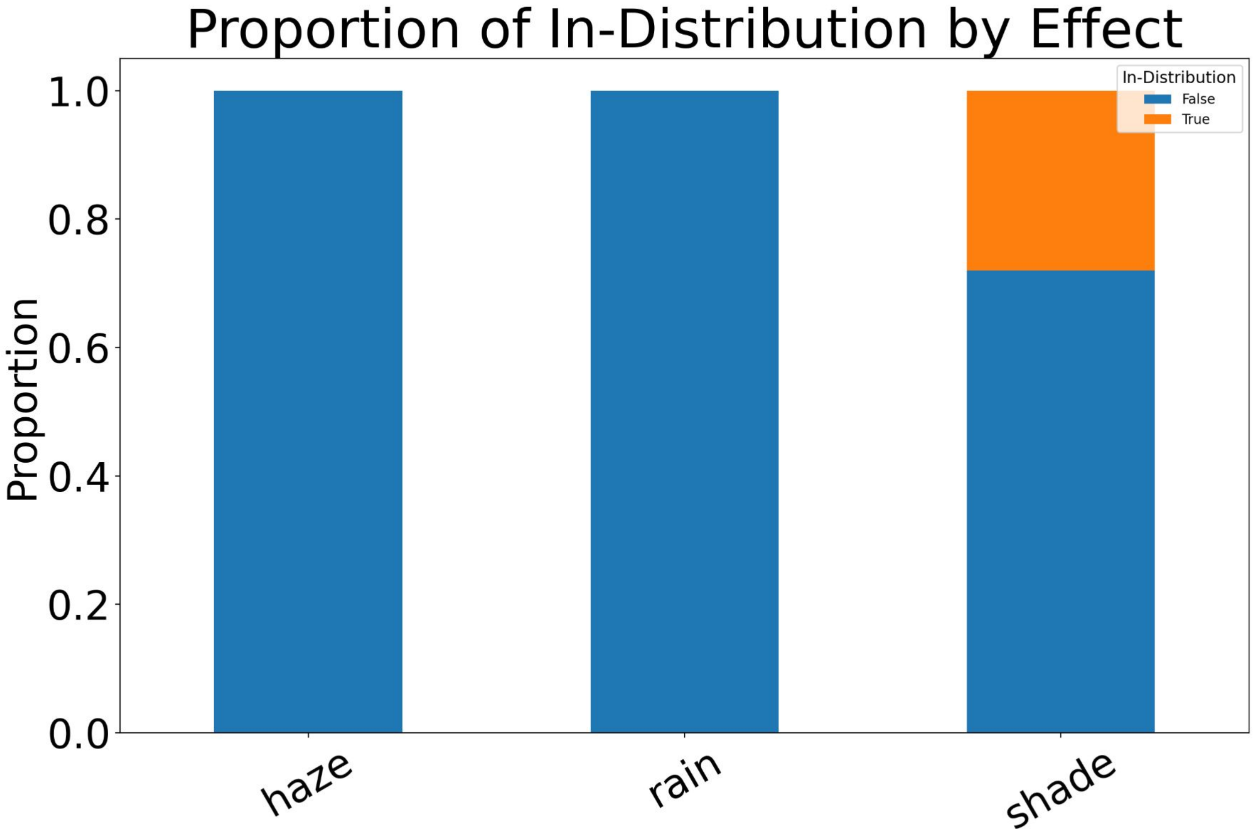
Does OOD detection work on modified images? (True Positive)

- Using a single sample camera input, tried 50 weather intensities on all weathers
- Result is presented on the next slides

OOD distribution single frame – Maha distance



OOD distribution single frame – Autoencoder



Fallback Method Experimentation Setup

- Testing scenario: lead slowdown
- Testing parameters:
  - 11 variations of the lead slowdown scenario
    - They do not contain accidents on their own
  - 3 weathers
  - 10 intensities of weather changes
- Running modes for comparison
  - Without weather modification to generate original images (11 configurations)
  - Baseline with modified input but using vanilla LBC (330 configurations)
  - With fallback method using only Mahalanobis Distance (330 configurations)
    - Why not Autoencoders

Results

Baseline

Event metric	Haze	Shade	Rain
Route diversion percentage	0	0	50
Route diversion degree	0	0	15.42
Vehicle collision percentage	100	10	10
Wall collision percentage	0	0	35.45

Maha

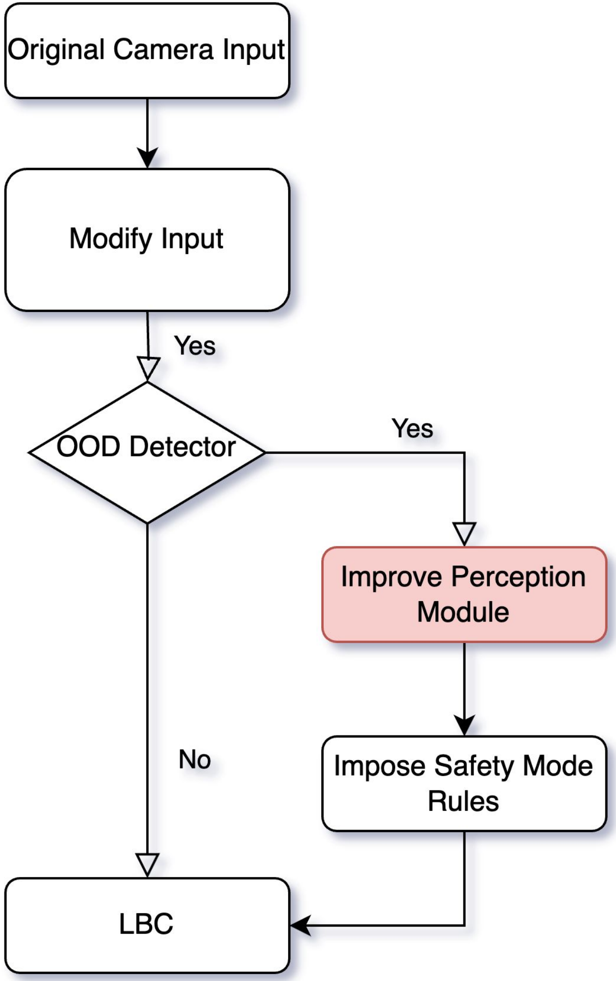
Event metric	Haze	Shade	Rain
Route diversion percentage	0	0	4.55
Route diversion degree	0	0	16.61
Vehicle collision percentage	9.09	0.91	1.82
Wall collision percentage	0	0	2.73

Discussion

- OOD-Maha is not good at identifying Hazed images, however Autoencoder is.
- Could experiment more to obtain more reasonable thresholds for determining OOD
- ODAF performs 11x better than baseline

Future Work

1. To improve robustness, could have a perception module to denoise the image.
2. Having a real fallback mode with a safer agent that specializes in parking the car on the road.



Contributions

- Jack
  - Testing and modifying OOD detection method and fallback mode
  - Integrating OOD detection method and fallback mode into pipeline
  - Designing and running experiments
  - Experiment analysis
- Barney
  - Implementation & testing of OOD detectors
  - OOD distribution results analysis & visualization
- Weihang
  - Implementation & testing of OOD detectors
  - OOD distribution results analysis
- Heramb
  - Implementation of image noise modification.
  - Implement automated pipeline to analyse results from testing