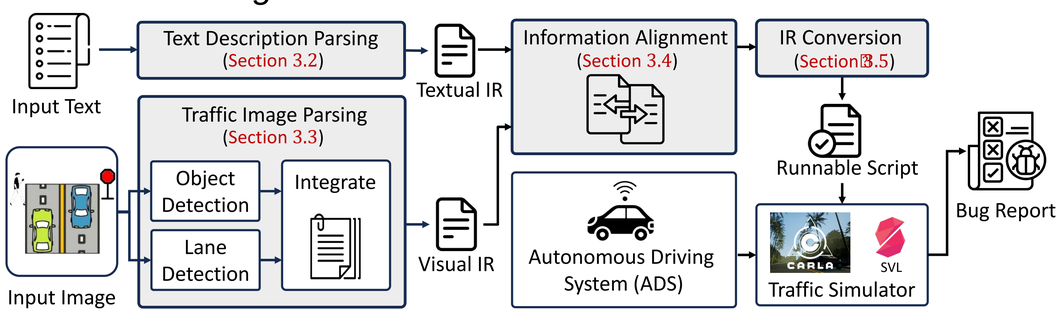
Multi-Modal Traffic Scenario Generation for Autonomous Driving System Testing

Zhi Tu and Tianyi Zhang, Purdue University
https://hcss.cs.purdue.edu/trafficcomposer

Motivation

- Manually constructing traffic scenarios for ADS testing is tedious and time-consuming.
- We propose TRAFFICCOMPOSER, a ***multi-modal*** approach to automate the generation of executable traffic scenarios in simulation.
- The generated scenarios can be used for direct ADS testing and also as high-quality seeds for ADS fuzzing or search-based testing.

Framework Design



- Text information extraction by an LLM-based textual description parser.
- Visual information extraction by pre-trained computer vision models for object and lane detection.
- A ***domain-specific language (DSL)*** is designed and used as an intermediate representation (IR) to encapsulate information from two modalities.

Role setting
You will work as an autonomous driving system testing engineer. Your task is to derive a test traffic scenario from the given traffic scenario description.
Breaks out of each step of the information parsing
Approach this task step-by-step, take your time, and do not skip steps. Step 1: Derive the '*weather*' element of the test scenario... Step 2: Derive the '*time*' of the test scenario... Step 3: Derive the test scenario's '*road_network*' element. The '*road_network*' element contains '*road_type*', '*traffic_sign*', '*traffic_light*' and '*lane_number*'... .....
Grammar of the DSL
Few-shot learning examples

Prompt design of the LLM-based text description parser

Multi-modal Input Example

It is currently daytime and cloudy.

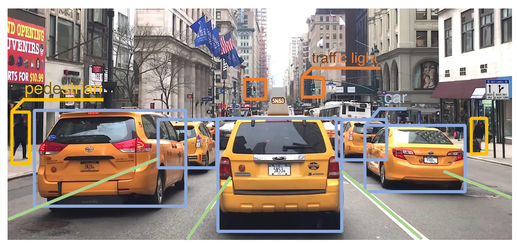
The ego vehicle is on the middle lane on a five-lane road.

There are pedestrians walking on both sides.

There is a orange car in front of the ego vehicle.

There are other cars driving in front of the ego vehicle in both left and right lanes.

The traffic light for the direction of the ego vehicle is green.



DSL Design

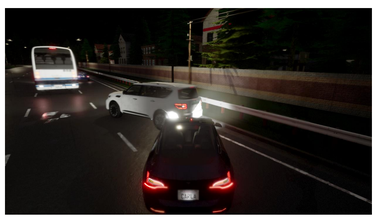
Scenario	::=	*Environment*; *Road_network*; *Actors*
Environment	::=	*weather*; *time*
weather	::=	`rainy` | `foggy` | `snowy` | `wet` | ...
time	::=	`daytime` | `nighttime`
Road_network	::=	*road_type*; *traffic_signals*; *lane_number*
road_type	::=	`intersection` | `roundabout` | ...
traffic_signals	::=	*traffic_signs*, *traffic_light*
traffic_signs	::=	ϵ | *traffic_sign*; *traffic_signs*
traffic_sign	::=	`stop_sign` | `speed_limit_sign` | ...
traffic_light	::=	ϵ |`red_light` | `green_light`
lane_number	::=	`0` | `1` | `2` | `3` | ...
Actors	::=	*ego_vehicle*; *npc_actors*
ego_vehicle	::=	*behavior*; *position*; *lane_idx*
npc_actors	::=	ϵ | *npc_actor*; *npc_actors*
npc_actor	::=	*actor_type*; *behavior*; *position*
actor_type	::=	`car` | `truck` | `train` | `pedestrian` | ...
behavior	::=	`go_forward` | `turn_left` | `static` | ...
position	::=	*reference_point*; *relative_position*
reference_point	::=	`ego_vehicle` | *road_type* | *traffic_sign*
relative_position	::=	`front` | `behind` | `left` | `on` | ...
lane_idx	::=	`0` | `1` | `2` | `3` | ...

Demonstrations of Exposed ADS Failures

Apollo runs a red light at an intersection due to tailgating the front car.

Autoware fails to yield to another vehicle and crashes while merging into the left lane.

MMFN collides with another vehicle while merging into the right lane to exit the road.



TransFuser collides with another vehicle while avoiding a stopped car in the rain.

Evaluation

Scenario Generation Accuracy

Method	Accuracy (%)
TARGET [1]	74.3 (±1.5)
gpt-4o-2024-05-13	89.7 (±5.2)
gpt-4o-mini-2024-07-18	79.1 (±6.1)
Claude-3-5-sonnet-20241022	86.1 (±4.3)
Claude-3-5-sonnet-20240620	84.1 (±4.9)
Llama-3.2-90B-Vision-Instruct	80.2 (±6.1)
Llava-v1.6-34b	72.2 (±4.2)
TRAFFICCOMPOSER	**97.0 (±1.2)**

Ablation study

Modality Used	Accuracy (%)
Textual input only	79.2(±1.1)
Visual input only	51.8(±0.0)
TRAFFICCOMPOSER	**97.0(±1.2)**

Effectiveness of TrafficComposer Scenarios in ADS Testing

Number of crashes or traffic rule violations detected in direct testing

ADS under test	Apollo	Autoware	MMFN	TransFuser	Roach	Behavior Agent
# failures	7	9	19	24	14	27

Average number of detected crashes and violations in fuzz testing

Initial Seed Setting	Apollo	Autoware	MMFN	TransFuser	Roach	Behavior Agent
original	22.1	19.2	22.6	20.9	23.7	36.0
TARGET	21.8	20.9	25.2	20.1	28.3	41.6
TRAFFICCOMPOSER	**39.1**	**41.6**	**43.8**	**45.1**	**48.3**	**55.2**

Average time consumed (in minutes) to detect the first crash or violation

Initial Seed Setting	Apollo	Autoware	MMFN	TransFuser	Roach	Behavior Agent
original	31.6	21.4	25.2	22.7	18.4	12.7
TARGET	27.8	21.8	18.7	21.4	19.3	10.4
TRAFFICCOMPOSER	**13.4**	**15.3**	**12.7**	**10.3**	**9.7**	**6.1**

[1] Deng, Y., Yao, J., Tu, Z., Zheng, X., Zhang, M., & Zhang, T. (2023). TARGET: Automated Scenario Generation from Traffic Rules for Testing Autonomous Vehicles. ArXiv. https://arxiv.org/abs/2305.06018

PURDUE UNIVERSITY®

Award ID#: 2416835