AutoMT: A Multi-Agent LLM Framework for Automated Metamorphic Testing of Autonomous Driving Systems - Supplementary Material

Anonymous Author(s)

1 Details of ontology elements

In this paper, we identified a total of 38 traffic rules from Germany and 72 from California. Due to space limitations, we do not list all the ontology elements in the paper. Instead, we summarize the extracted ontology elements in Tables 1 and 2, corresponding to California and Germany respectively. These tables follow the same subcategory structure as paper. Since the expected behavior of the ego-vehicle remains consistent across all rules, it is omitted from the these two tables for brevity.

Road Type
All roads, Intersections, Crosswalks, School Zones, Unmarked Crosswalk, Work Zone, Alleys, On-ramps, Off-ramps, Freeways, Railroad Crossings, Roundabouts
Manipulation (Adds)
Red Light, Yellow Light, Green Light, Red Arrow, Green arrow, STOP Sign, YIELD Sign, 5-Sided Sign, Warning Signs, Crosswalk Markings, Limit lines, Bike Lanes, Center Left Turn Lanes, Flashing Red Light, Flashing Yellow Light, No U-Turn Sign, Wrong Way Sign, Do Not Enter Sign, Speed Limit Signs, High-Occupancy Vehicle (HOV) Lanes, Ramp Meter Signals, Shared Lane Markings (Sharrows), Solid and Broken Yellow Lines (as passing/no-passing indicators), Reflective Road Markers, Railroad Crossing Sign (Crossbuck), Pedestrian Signal Indicators (WALK/DON'T WALK), Raised Pavement Markers, Traffic infrastructure, Vehicle, Emergency vehicle, Tow truck, Road work vehicle, Pedestrian, Person using roller skates, Person using a skateboard, Person with a disability using a wheelchair, Person with a disability using a tricycle, Person with a disability using a quadricycle, Child, Senior (elderly person), Person with small children, Bicyclist, Heavy Traffic, Motorcyclist, Animals, deer, dogs, Traffic Officer, Flaggers (in construction zones), Escorted School Children, School Patrol, Crossing Guard, School bus, collision, Livestock, Obstacle
Manipulation (Replaces)
Rain, Snow, Mud, Ice, Wet road, Fog, Heavy smoke, High winds, Low lighting (implied), Sun Glare, Flooded Roads, Dust Storms, Black Ice, Potholes, Loose Gravel, Drizzle, Slippery road

Table 1: Complete ontology elements extracted from California traffic rules

Road Type
All roads, Intersection, Port Area, Field path, Forest path, Footpath, Bicycle path, Industrial area, Residential area, Pedestrian zone, Tunnel, Level crossing, Motorway, Hard shoulder
Manipulation (Adds)
St. Andrew's Cross sign, Bus Stop Sign, Start of 30 km/h Zone Sign, Maximum Speed Limit Sign, No Entry Sign, Prohibition for Motor Vehicles Sign, Prohibition for Vehicles of All Kinds Sign, Warning Signs, Right-of-Way Regulation Signs, Stop sign, Yield sign, Pedestrian crossing marking, Lane markings (solid, broken, directional arrows), Traffic island, Reflective posts, Traffic light (Signal system), Guardrail, Chevron signs, Distance markers, Overhead signage, Pedestrian, Bicycle, Bicycle with auxiliary motor, Electric micro-vehicle, Passenger, Driver in front, Vehicle behind, Railway employee with flag, Obstacle on the road, Tractor, Horse-drawn vehicle, Trailer, Animal, Dog, Deer, Child, Blind pedestrian, Person with limited mobility, Skater/scooter rider, Police officer (directing traffic), Construction worker, Overhanging load, Vehicle convoy, Road narrowing, Emergency vehicle, Stopped vehicle, Oncoming vehicle, Turning vehicle, Rail vehicle, Public transport bus, School bus, Line bus, Motor vehicle, Multi-track motor vehicle
Manipulation (Replaces)
Fog, Snowfall, Rain, Black ice, Strong wind, Glare from sun, Poor visibility (general), Night driving (darkness)

Table 2: Complete ontology elements extracted from German traffic rules

2 IR-related Metrics

To evaluate the performance of RAG, we adopt three metrics [3]: Recall@K, Precision@K, and F1 Score.

Recall@K measures the proportion of relevant instances that have been retrieved among the total number of relevant cases, considering only the top-k results:

$$\text{Recall@K} = \frac{|RD \cap Top_k^d|}{|RD|} \tag{1}$$

where, RD denotes the set of relevant documents, Top_k^d represents the top-k retrieved documents.

Precision@K measures the fraction of relevant instances among the retrieved instances, considering only the top-k results:

$$\text{Precision@K} = \frac{TP}{TP + FP} \tag{2}$$

where, TP is the number of true positives, and FP is the number of false positives.

F1 Score measures the balance between precision and recall, defined as the harmonic mean of the two:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

We computed the RAG metrics with top-k set to 20, and the results are summarized in Table 3. In LangChain RAG frameworks, the function vector_store.as_retriever() returns a Retriever object, which is responsible for fetching the most relevant documents from the vector database. In our implementation, we configured the retriever as vector_store.as_retriever(search_kwargs="k":20), where the parameter k specifies the number of documents retrieved.

Scenario	Recall@K	Precision@K	F1 Score
California	0.5032	0.9054	0.6375
Germany	0.7471	0.6595	0.6987
Average	0.6410	0.7664	0.6721

Table 3: Recall, Precision, and F1 Score results for California, Germany, and their average across all test cases.

3 Details of baseline Auto MT Pipeline w/o Traffic Rules

Table 4 illustrates the detailed prompt used in the baseline Auto MT Pipeline w/o Traffic Rules.

Role Setting

You are an expert in traffic rules and scene understanding.

Prompt

Metamorphic Testing (MT) is a method used in autonomous vehicle testing. It defines how the behavior of a system should change when its inputs are changed in a specific way. Given an image of a traffic scene, generate a Metamorphic Relation (MR) in the format:
Given the ego-vehicle approaches to [Road Type]
When method [Manipulation]
Then ego-vehicle should [Ego-Vehicle Expected Behavior]
The [Ego-Vehicle Expected Behavior] must be one of the following four actions:
- slow down, - keep current, - turn left, - turn right
Additional context about the ego-vehicle: {vehicle_info}
Example:
Given the ego-vehicle approaches to an intersection
When method adds a steady red light on the roadside
Then ego-vehicle should slow down
Now generate an MR for the image below. Only output the MR without explanation
User: One Image or Video.

Table 4: Prompt for Auto MT Pipeline w/o Traffic Rules

4 Details of baseline Auto MT Pipeline w/ Traffic Rules

Table 5 illustrates the detailed prompt used in the baseline Auto MT Pipeline w/ Traffic Rules.

5 Details of Computational Cost

In this work, we used ChatGPT to read PDFs and summarize traffic rules. Since this part was implemented via web ChatGPT [6] and human evaluation, it is not included in the computation overhead.

Role Setting

You are an expert in traffic rules and scene understanding.

Prompt

Metamorphic Testing (MT) is a method used in autonomous vehicle testing.
Traffic rules: It defines how the behavior of a system should change when its inputs are changed in a specific way.
{Traffic Rules}: All traffic rules of this region. e.g. If a pedestrian makes eye contact with you, they are ready to cross the street. Yield to the pedestrian.
Given an image of a traffic scene and traffic rules, extract a Metamorphic Relation (MR) from traffic rule in the format:
Given the ego-vehicle approaches to [Road Type]
When method [Manipulation]
Then ego-vehicle should [Ego-Vehicle Expected Behavior]
The [Ego-Vehicle Expected Behavior] must be one of the following four actions:
- slow down, - keep current, - turn left, - turn right
Additional context about the ego-vehicle: {vehicle_info}
Example:
Given the ego-vehicle approaches to an intersection
When method adds a steady red light on the roadside
Then ego-vehicle should slow down
Now extract an MR from traffic rules for the image below. Only output the MR without explanation.
User: One Image or Video.

Table 5: Prompt for Auto MT Pipeline w/ Traffic Rules

First, M-Agent consumed an average of about 4.8k tokens. Three LLMs were used for MR generation: GPT (2.1837s), Claude (3.9257s), and Qwen (2.082s). selfcheckGPT took 4.845s. Since the three models can run in parallel, the overall runtime of M-Agent was 8.7707s.

Second, the baseline runtime was 2.820s with 2.1k tokens for the AutoMT Pipeline w/o Traffic Rules. The baseline runtime was 3.1348s with 3.4k tokens for the AutoMT Pipeline w/ Traffic Rules. Considering the high cost of directly inputting the full PDF, we instead extracted and fed the traffic rules into the AutoMT Pipeline w/ Traffic Rules.

At the same time, we used RAG and T-Agent for scenario matching MRs. T-Agent cost 3.67s and RAG cost 11.49s. Token consumption at this stage was about 4200, and the runtime was 15.16s.

The same methods as the baseline were used. We used an RTX 3090 GPU for F-Agent. Int4-flux.1-fill-dev cost 6.9616s, and instruct-pix2pix cost 2.07s. Vista, due to its 40GB VRAM requirement, was run with a quantized model with CPU–GPU offloading, averaging 84.33s. As shown in Table 6, the computation time overhead of each component is summarized.

6 Details of baseline Auto MT with Manually Defined MRs

We selected nine manually extracted Metamorphic Relations (MRs) from previous research, each with a clear correspondence to formal traffic rules. As these MRs were originally expressed in inconsistent

Method	MR Generation	Test cast Matching MR	Test Case Generation	Total
AutoMT	8.07s	15.16s	91s	114.23s
AutoMT Pipeline w/o Traffic Rules	–	2.82s	91s	93.82s
AutoMT Pipeline w/ Traffic Rules	–	3.13s	91s	94.13s

Table 6: Computation cost of different pipelines.

formats, we standardized them into the "Given–When–Then" structure used in our paper. The table 7 presents the selected MRs, along with their unified format and references to the original sources.

Details of baseline Auto MT with Manually Defined MRs
MR1 : Given the ego-vehicle approaches to any roads, When method adds a pedestrian on the roadside, Then ego-vehicle should slow down.
MR2 : Given the ego-vehicle approaches to any roads, When method adds a speed limit sign on the roadside, Then ego-vehicle should slow down.
MR3 : Given the ego-vehicle approaches to any roads, When method replaces time into night, Then ego-vehicle should slow down.
MR4 : Given the ego-vehicle approaches to any roads, When method replaces weather into snow, Then ego-vehicle should slow down.
MR5 : Given the ego-vehicle approaches to any roads, When method replaces sunny with rainy, Then ego-vehicle should slow down.
MR6 : Given the ego-vehicle approaches to any roads, When method adds a vehicle on the road, Then ego-vehicle should slow down.
MR7 : Given the ego-vehicle approaches to any roads, When method adds a cyclist on the road, Then ego-vehicle should slow down.
MR8 : Given the ego-vehicle approaches to any roads, When method adds a red light on the roadside, Then ego-vehicle should slow down.
MR9 : Given the ego-vehicle approaches to any roads, When method adds a green light on the roadside, Then ego-vehicle should keep current.

Table 7: MR of Auto MT with Manually Generated MR

7 Details of ADSs

All autonomous driving systems (ADS) are independently trained on both the German A2D2 dataset and the Californian Udacity dataset for two prediction tasks: steering angle and vehicle speed. All models are implemented in PyTorch and trained using the Adam optimizer with a learning rate of 0.001 for 30 epochs. A batch size of 128 is used during training. The loss function is the Mean Absolute Error (MAE), implemented using the L1Loss criterion. The model checkpoint with the lowest validation loss is selected for evaluation. As shown in table 8 and Table 9, the results summarize the mean absolute error (MAE) of the trained autonomous driving system (ADS) models on the A2D2 and Udacity datasets.

The detailed configurations of each model are summarized as follows:

PilotNet [1] is a convolutional neural network designed for end-to-end autonomous driving. It takes a single RGB image of size 3×160×320 as input and processes it through five convolutional layers to extract visual features, followed by fully connected layers for regression. The architecture can be summarized as: Conv(5×5, 3⟶24) ⟶ Conv(5×5, 24⟶36) ⟶ Conv(5×5, 36⟶48) ⟶ Conv(3×3, 48⟶64) ⟶ Conv(3×3, 64⟶64) ⟶ Flatten ⟶ FC(1200) ⟶ FC(100) ⟶ FC(50) ⟶ FC(10) ⟶ Output(1). The final output is a single value representing either speed or steering.

Model	Speed MAE	Steering MAE
Resnet101	3.7668	0.5842
Vgg16	2.3423	0.5613
Epoch	2.8702	0.5412
PilotNet	3.0833	0.5548
CNN_LSTM	3.0252	0.7571
CNN_3D	**1.6100**	**0.3622**

Table 8: Prediction MAE of different models on the A2D2 dataset

Model	Speed MAE	Steering MAE
Resnet101	**3.0960**	0.7543
Vgg16	4.4993	0.6570
Epoch	3.3397	0.5593
PilotNet	3.6098	0.7889
CNN_LSTM	3.3159	0.9125
CNN_3D	3.8554	**0.6296**

Table 9: Prediction MAE of different models on the Udacity dataset

Epoch [4] is a CNN-based model for end-to-end driving tasks. It takes a 3×160×320 RGB image as input and processes it through four convolutional layers with 3×3 kernels, padding=1, batch normalization, ReLU activation, and 2×2 max pooling. The resulting feature map is flattened and passed through a fully connected layer with 256 units and dropout. The architecture is summarized as: Conv(3×3, padding=1, 3⟶32) ⟶ Conv(3×3, padding=1, 32⟶64) ⟶ Conv(3×3, padding=1, 64⟶128) ⟶ Conv(3×3, padding=1, 128⟶256) ⟶ Flatten(⟶51200) ⟶ FC(51200⟶256) ⟶ FC(256⟶1). The final output is a single value representing either speed or steering, depending on the task mode.

ResNet101-based ADS [2] is a deep regression model based on a ResNet101 backbone pretrained on ImageNet. It takes a 3×160×320 RGB image as input and processes it through the ResNet101 convolutional layers. The original classification head is replaced with a batch normalization layer, dropout (p=0.25), and a linear layer that reduces the 2048-dimensional pooled feature to 256 dimensions, followed by ReLU activation. The architecture is summarized as: ResNet101 (with AdaptiveAvgPool) → FC(2048→256) → FC(256→1). The final output is a single value representing either speed or steering, depending on the task.

VGG16-based ADS [2] is a convolutional regression model for end-to-end driving. It takes a 3×160×320 RGB image as input and passes it through the convolutional backbone of VGG16, followed by an additional 3×3 convolutional layer and adaptive average pooling. The resulting feature map is flattened into a 25,088-dimensional vector and processed through a fully connected layer. When state inputs are used, their 256-dimensional encoding is concatenated with the image features before the final prediction. The architecture is summarized as: VGG16 (features) → Conv(3×3, 512→512, padding=1) → AdaptiveAvgPool(7×7) → Flatten(→25088) → FC(25088→256) → FC(256 or 512→1). The model outputs a single value representing either speed or steering, depending on the task.

The other two ADSs process multiple consecutive frames:

CNN-LSTM [5] is a hybrid sequential model for end-to-end driving. The model takes a sequence of four RGB images (each of size 3×160×320) as input. Each frame is individually processed through a shared CNN encoder consisting of five convolutional layers with batch normalization and ELU activation. The resulting feature maps are downsampled using adaptive average pooling to 2×2 and flattened into 256-dimensional vectors. These are concatenated into a temporal sequence of shape 4×256, which is passed into an LSTM with 100 hidden units. The last LSTM output is used for final regression. The architecture is summarized as: Conv(5×5, 3→24, stride=2) → Conv(5×5, 24→36, stride=2) → Conv(5×5, 36→48, stride=2) → Conv(3×3, 48→64) → Conv(3×3, 64→64) → AdaptiveAvgPool(2×2) → Flatten(→256) → LSTM → FC(100→50) → FC(50→10) → FC(10→1). The final output is a single value representing either vehicle speed or steering angle, depending on the task.

CNN3D [5] is a spatiotemporal convolutional model for end-to-end driving. It takes a sequence of four RGB images (each of size 3×160×320) and processes them as a 5D tensor with shape 3×4×160×320. The input is passed through five 3D convolutional layers with batch normalization and ELU activation: Conv3D(3→24, 2×5×5, stride=1×2×2) → Conv3D(24→36, 2×5×5, 1×2×2) → Conv3D(36→48, 2×5×5, 1×2×2) → Conv3D(48→64, 1×3×3) → Conv3D(64→64, 1×3×3) → AdaptiveAvgPool3D(3×3×3) → Flatten(→1728) → FC(1728→256) → FC(256→50) → FC(50→10) → FC(10→1).The final output is a single value representing either vehicle speed or steering angle, depending on the task.

8 Details of evaluation metrics

As shown in Table 10 and Table 11, the prompts used for **Scenario Alignment** and **Logical Alignment** in the paper are presented respectively.

Role Setting

You are a visual reasoning AI specialized in driving scenes.

Prompt

You will be shown two images.
The given scenario description is: {scene_description}.
Determine whether **both** images match this scenario.
Focus on key driving elements like road types, lane markings, vehicles, traffic signs, and pedestrians.
Minor visual differences are acceptable, but both images must clearly represent the described scenario.
If the description is generic (e.g., "all roads"), reply "Yes" unless the image is clearly unrelated.
Reply only with "Yes" if both images reasonably match the scenario description, or "No" otherwise.
User: {original test case}
User: {follow test case}

Table 10: Prompt for Scenario Alignment

Role Setting

You are a reasoning assistant for validating the logical consistency and behavioral correctness of metamorphic relations (MRs) in autonomous driving scenarios.

Prompt 1: Logical Consistency Check

Your task is to determine whether the ego-vehicle's expected behavior is clearly impossible.
Example:
- If the MR says "turn left" but there is no turn-related element (e.g., sign, road), respond "No".
- If the MR adds a "must turn left" sign and the expected behavior is "turn left", respond "Yes".
Respond only "Yes" or "No" based on the MR. If the answer is "No", briefly explain why.
User: {MR}

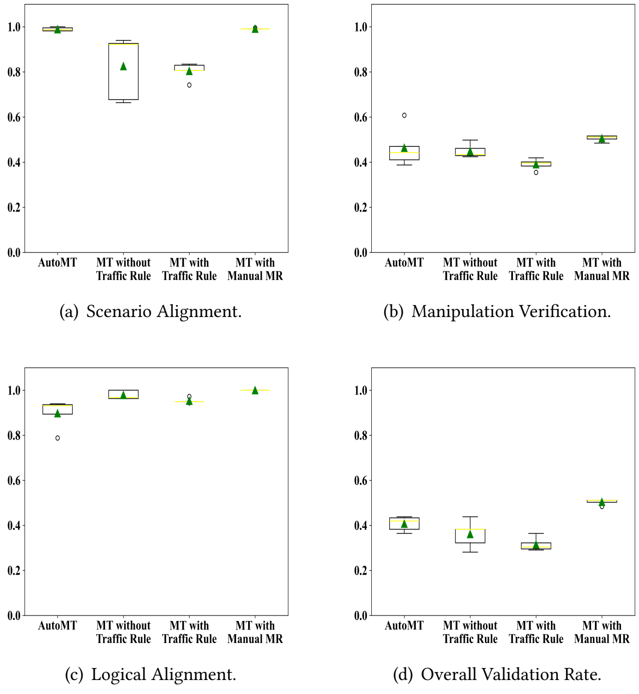
Prompt 2: Behavior Rationality Check with Speed Context

You are given a metamorphic relation (MR) and the current ego-vehicle speed.
The expected behavior "slow down" is considered unreasonable if the MR adds a speed limit sign that is higher than or equal to the current speed.
Respond only "Yes" or "No". If the answer is "No", briefly explain why.
`Current speed of the ego-vehicle: {speed} km/h`
User: {MR}

Table 11: Prompts for Logical Alignment

9 RQ 2 SUPPLEMENTARY MATERIAL

As shown in Fig 1 and Fig 2, the boxplots for the validation rate of all models in Germany and California are presented. From these boxplots, we can observe that AutoMT yields consistent results. First, in the Scenario Alignment task, AutoMT achieved results that were second only to those of MT with Manual MR, consistent with the analysis presented in the paper. The Manipulation Verification results of AutoMT showed some fluctuations across both regions, though the mean values are representative of the AutoMT overall performance. For the Logical Alignment results of AutoMT, we observed that a Logical Alignment in Germany is outlier, which caused the mean result of AutoMT to be lower. We believe the likely cause for this is the hallucination of large language models (LLMs). During the RAG process, to enhance the LLM's ability to match different MRs, we set a high TopK value for the entire RAG retriever. This led to the LLM receiving a large number of tokens, which can lead to the inclusion of irrelevant or non-optimal information instead of correctly analyzing the speed limit itself. Lastly, we can bserve that our average performance outperform MT without Traffic Rule (Auto MT Pipeline w/o Traffic Rules) and MT with Traffic Rule (Auto MT Pipeline w/ Traffic Rules). Additionally, we can observe that our average performance is slightly lower than MT with Manual MRs (Auto MT with Manually Defined MRs). Through confidence interval analysis across five validation rounds, our method achieved a validation rate confidence interval of (0.4207, 0.4588), which substantially outperforms MT without Traffic Rule (0.2861, 0.4031) and MT with Traffic Rule (0.2310, 0.2971). While our method's performance is slightly lower than the alternative MT with Traffic Rule implementation (0.4347, 0.4746), the overlapping confidence intervals suggest that the performance differences are not statistically significant.



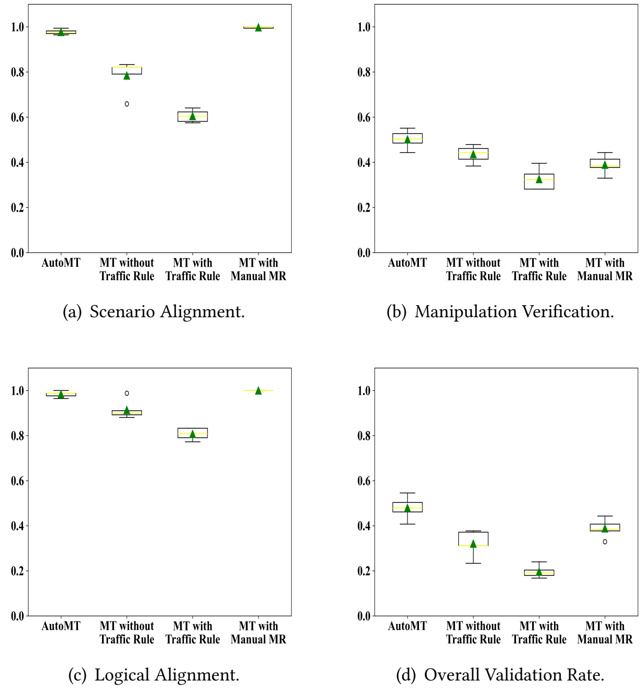
(a) Scenario Alignment.

(b) Manipulation Verification.

(c) Logical Alignment.

(d) Overall Validation Rate.

Figure 1: Boxplots of Validation Rates in German.



(a) Scenario Alignment.

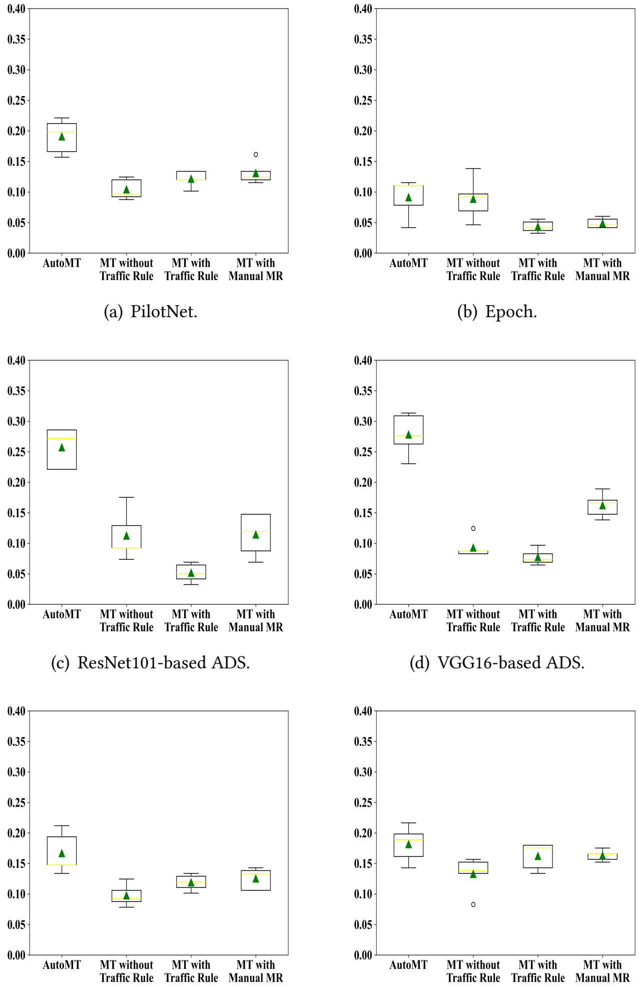
(b) Manipulation Verification.

(c) Logical Alignment.

(d) Overall Validation Rate.

Figure 2: Boxplots of Validation Rates in California.

As shown in Fig 3 and Fig 4, the boxplots for the violation rate of all models in Germany and California are presented. From the subplots, we can see that AutoMT achieved high violation rate in each round of experiments, making the conclusion consistent with the analysis presented in the paper.



(a) PilotNet.

(b) Epoch.

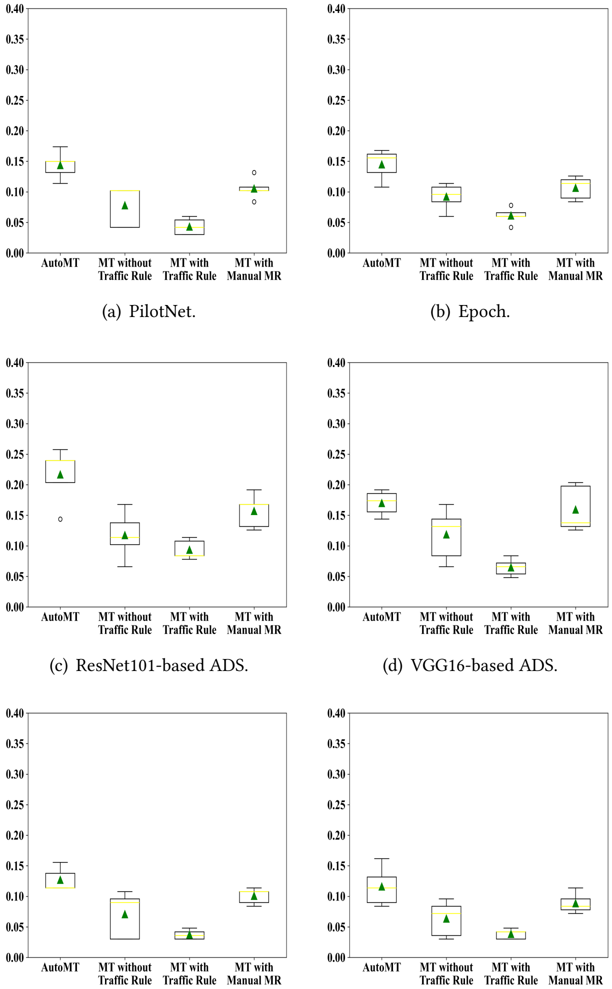
(c) ResNet101-based ADS.

(d) VGG16-based ADS.

(e) CNN-LSTM.

(f) CNN3D.

Figure 3: Boxplots of Violation Rates in German.



(a) PilotNet.

(b) Epoch.

(c) ResNet101-based ADS.

(d) VGG16-based ADS.

(e) CNN-LSTM.

(f) CNN3D.

Figure 4: Boxplots of Violation Rates in California.

10 RQ 3&4 SUPPLEMENTARY MATERIAL

All traffic rule is in two regions are shown in Table 12 and Table 13. The boxplot of all human assessment results of realistic of AutoMT of two regions are shown in Fig 5 and Fig 6. Each test case in these Figs is labeled in the format “RA-B”, where “A” denotes the traffic rule number and “B” indicates the index of the follow-up test case generated from that rule. The boxplot of all human assessment results of reasonability of ADS’ prediction in two regions are shown in Fig 7 and Fig 8. In the box plot, yellow represents the median, green triangles represent the mean. In Fig 7 and Fig 8, we mark cases that violate MR as "violation" and display them in red.

Traffic Rules from Germany

1. Special consideration must be given to pedestrians; if necessary, one must wait.
2. A red light signals to stop.
3. If visibility is reduced to less than 50 meters due to snowfall, the speed must not exceed 50 km/h, unless a lower speed is required.
4. If visibility is reduced to less than 50 meters due to fog, the speed must not exceed 50 km/h, unless a lower speed is required.
5. When passing public transport buses that are stopped at bus stops with their hazard lights on, vehicles must only pass at walking speed and maintain a distance that ensures the safety of passengers is not compromised.
6. Maximum Speed Limit Sign: Command or Prohibition.
7. Anyone driving a vehicle must behave in such a way to assist people in need, particularly by reducing speed and being prepared to brake.
8. The distance to a vehicle ahead must generally be large enough so that one can still stop behind it if it suddenly brakes.
9. At a road narrowing, an obstacle on the road, or a stopped vehicle, anyone who wants to pass on the left must let oncoming vehicles go through.
11. Where public transport buses, trams, and designated school buses are stopping, vehicles, including those in the oncoming traffic, may only pass cautiously.
12. Rain can make the roads slippery. Drive more slowly than you would on a dry road.
13. STOP Sign — Give way.
14. The distance to a vehicle ahead must generally be large enough so that one can still stop behind it if it suddenly brakes.
15. Prohibition for Vehicles of All Kinds sign: Command or Prohibition.
16. If the railroad crossing cannot be crossed swiftly and without delay due to road traffic.
17. Must let oncoming vehicles pass, including bicycles even if they are traveling on or alongside the roadway in the same direction.

Table 12: The traffic rule in human evaluation (by Germany)

Traffic Rules from California

1. Snow can make the roads slippery. Drive more slowly than you would on a dry road.
2. A red traffic signal light means STOP.
3. A yellow traffic signal light means CAUTION.
4. Slow down and be ready to stop to let any pedestrian pass before you proceed.
5. Yield to emergency vehicles.
6. It is best to avoid driving in heavy fog or smoke. Consider postponing your trip until the fog clears.
7. When a vehicle with one light drives toward you, drive as far to the right as possible. It could be a motorcyclist.
8. When a person traveling on something other than a vehicle or bicycle (e.g., roller skates, skateboard) crosses the roadway with or without a crosswalk, you must use caution and reduce your speed.
9. You must drive slower when there is heavy traffic.
10. Pedestrians using guide dogs or white canes have the right-of-way at all times. These pedestrians are partially or totally blind. Be careful when you are turning or backing up.
11. When approaching a stationary emergency vehicle with flashing emergency signal lights (hazard lights), move over and slow down.
12. Rain can make the roads slippery. Drive more slowly than you would on a dry road.
13. Some school buses flash yellow lights when preparing to stop to let children off the bus. The yellow flashing lights warn you to slow down and prepare to stop.
14. Drivers must move over and slow down for road work vehicles.
15. Stop sign: Make a full stop before entering the crosswalk or at the limit line.
16. When approaching a stationary emergency vehicle with flashing emergency signal lights (hazard lights), move over and slow down.
17. Slow down and be ready to stop to let any pedestrian pass before you proceed.
18. Go through the work zone carefully by: Slowing down.
19. A green traffic signal light means GO.
20. Slowing down. Stationary emergency vehicles or tow trucks displaying flashing amber warning lights.

Table 13: The traffic rule in human evaluation (by California)

References

[1] Mariusz Bojarski. 2016. End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316 (2016).

[2] Yao Deng, Xi Zheng, Tianyi Zhang, Huai Liu, Guannan Lou, Miryung Kim, and Tsong Yueh Chen. 2022. A declarative metamorphic testing framework for autonomous driving. IEEE Transactions on Software Engineering (2022).

[3] Aoran Gan, Hao Yu, Kai Zhang, Qi Liu, Wenyu Yan, Zhenya Huang, Shiwei Tong, and Guoping Hu. 2025. Retrieval Augmented Generation Evaluation in the Era of Large Language Models: A Comprehensive Survey. arXiv preprint arXiv:2504.14891 (2025).

[4] Chris Gundling. 2017. cg23. https://bit.ly/2VZYHGr.

[5] Zhihui Lai and Thomas Bräunl. 2023. End-to-end learning with memory models for complex autonomous driving tasks in indoor environments. Journal of Intelligent & Robotic Systems 107, 3 (2023), 37.

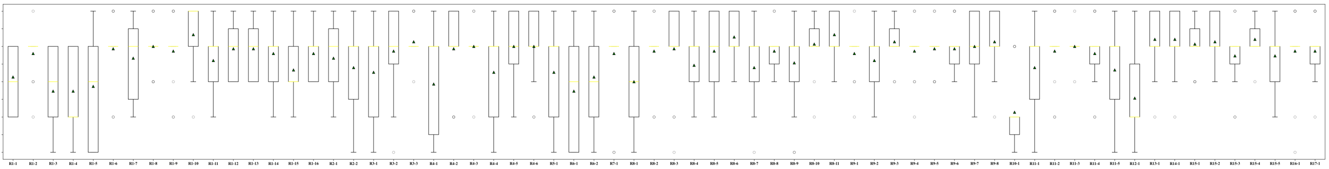


Figure 5: Human assessment results of realistic of follow-up video test cases in German.

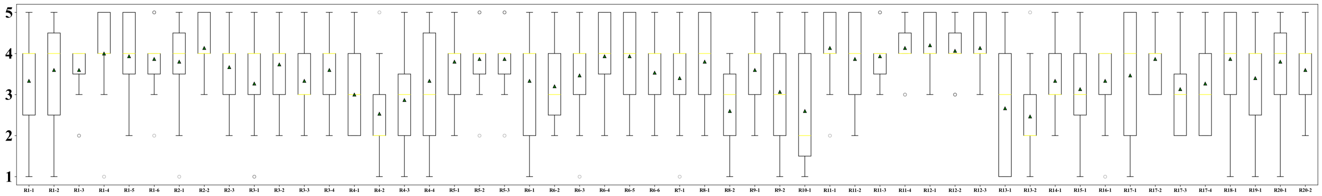


Figure 6: Human assessment results of realistic of follow-up video test cases in California.

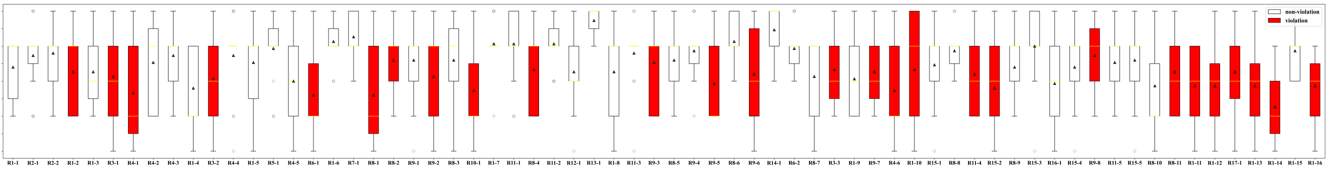


Figure 7: Human assessment of the reasonability of new prediction in German.

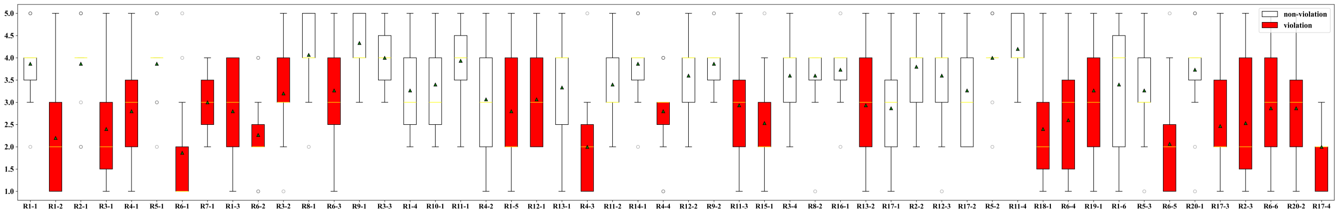


Figure 8: Human assessment of the reasonability of new prediction in California.

[6] Seung Yeob Shin, Fabrizio Pastore, Domenico Bianculli, and Alexandra Baicoianu. 2024. Towards Generating Executable Metamorphic Relations Using Large Language Models. arXiv preprint arXiv:2401.17019 (2024).