

A Digital Twin Framework for Metamorphic Testing of Autonomous Driving Systems Using Generative Model

Tony Zhang*, Burak Kantarci*, Umair Siddique**

*School of Electrical Engineering and Computer Science, University of Ottawa, Canada
{yzhan117, burak.kantarci}@uOttawa.ca

**reasonX Labs Inc., Ottawa, Canada
umair@reasonx.ai

Abstract—Ensuring the safety of self-driving cars remains a major challenge due to the complexity and unpredictability of real-world driving environments. Traditional testing methods face significant limitations, such as the oracle problem, which makes it difficult to determine whether a system’s behavior is correct, and the inability to cover the full range of scenarios an autonomous vehicle may encounter. In this paper, we introduce a digital twin-driven metamorphic testing framework that addresses these challenges by creating a virtual replica of the self-driving system and its operating environment. By combining digital twin technology with AI-based image generative models such as Stable Diffusion, our approach enables the systematic generation of realistic and diverse driving scenes. This includes variations in weather, road topology, and environmental features, all while maintaining the core semantics of the original scenario. The digital twin provides a synchronized simulation environment where changes can be tested in a controlled and repeatable manner. Within this environment, we define three metamorphic relations inspired by real-world traffic rules and vehicle behavior. We validate our framework in the Udacity self-driving simulator and demonstrate that it significantly enhances test coverage and effectiveness. Our method achieves the highest true positive rate (0.719), F1 score (0.689), and precision (0.662) compared to baseline approaches. This paper highlights the value of integrating digital twins with AI-powered scenario generation to create a scalable, automated, and high-fidelity testing solution for autonomous vehicle safety.

Index Terms—Digital Twin, Metamorphic Testing, Autonomous Driving Systems, Generative Version Model.

I. INTRODUCTION

The complexity of real-world situations makes the development and certification of autonomous driving systems (ADS) [1] extremely difficult. ADS must function in dynamic, unexpected contexts that necessitate the use of complex testing procedures, in contrast to standard software systems with clearly defined inputs and outputs. This issue becomes even more challenging due to the fact that many modern ADS architectures operate as black-box systems [2], which conceals their decision-making processes and makes it difficult to comprehensively validate their outcomes.

A possible method for verifying ADS, especially in situations without a conclusive test oracle, is Metamorphic Testing (MT) [3] in combination with a Digital Twin [4]. By analyzing

invariant relations between outputs when inputs go through regulated transformations, MT assesses the behavior of the system [5] and Digital Twin provides a dependable platform for consistent and repeatable testing [6]. However, traditional MT methods frequently depend on simple transformations and restricted metamorphic connections [7]. This simplicity can lead to systems that adapt to specific test patterns rather than developing genuine robustness, potentially resulting in models that perform adequately under test conditions but fail to generalize to real-world scenarios [8].

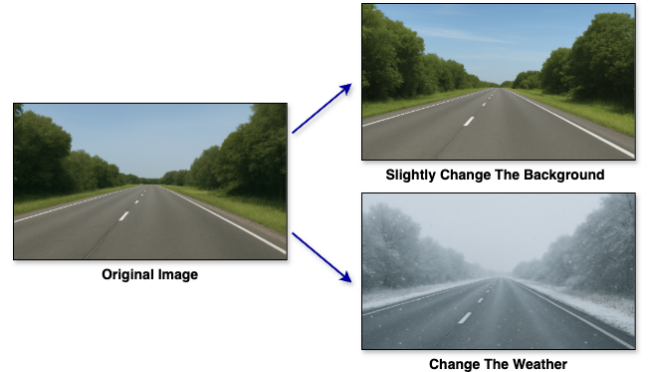


Fig. 1. The case of using a generative model to apply transformations to a real image.

To address these limitations, we propose incorporating advanced generative models, particularly Stable Diffusion [9]. While preserving environmental coherence, these generative models are excellent at creating Digital Twin and generating complex, controlled variations from input data [10]. We generate complex scenario variations—such as changes in lighting, weather, and lane layout—within the MT framework to fully assess ADS decisions. Using the Stable Diffusion-XL model, we subtly alter backgrounds while preserving the original lane direction in Figure 1, creating diverse metamorphic test cases.

Our methodology creates reusable Digital Twins based on ADS behavioral invariants and generates consistent but varied scenarios to increase test reproducibility. Path planning and obstacle detection are two important ADS components that

we assess using generative transformations to create a Digital Twin on camera inputs [11]. By overcoming the limitations of traditional MT and leveraging modern generative models, our framework achieves an F1 score of 0.689 and a precision of 0.662 significantly outperforming baseline methods such as CAE and SAE.

II. BACKGROUND

To ensure ADS safety, prior work has applied metamorphic testing and uncertainty-aware methods to detect anomalies in rare or uncertain scenarios. Digital twin frameworks have gained traction for enhancing runtime safety and anomaly detection in autonomous driving systems (ADS). Kumar et al. (2025) introduced the use of digital twins to assess the feasibility of operational failure detection and decision-making in autonomous vehicles [12]. Similarly, the Auto-DRIVE ecosystem by Samak et al. (2023) offers a modular, flexible, and open-access digital twin platform that enables hardware–software co-simulation, real-time testing, and scalable validation for both research and educational purposes in autonomous driving [13]. Wu et al. (2021) proposed a digital twin-enabled reinforcement learning framework that integrates a predictive environment model and rollout-compatible training to improve data efficiency and performance in autonomous driving control tasks [14]. Stocco et al. (2020) introduced SelfOracle, a runtime monitoring technique that estimates DNN confidence using autoencoders and time-series anomaly detection to forecast potential misbehaviors, achieving a 77% detection rate [15]. However, it lacks domain-specific metamorphic reasoning and may not generalize to unseen scenarios. Wang et al. (2023) proposed uncertainty-aware metamorphic testing for object detection, using Bayesian uncertainty to evaluate robustness under realistic perturbations, though it remains limited to offline use [16]. Most recently, Ayerdi et al. (2024) developed MarMot, a runtime monitoring approach that leverages domain-specific metamorphic relations to detect both external and internal anomalies in ADSs, outperforming prior methods like SelfOracle and MC Dropout in both simulated and real-world settings [17].

Prior studies improve DNN-based ADS robustness with metamorphic or uncertainty-aware methods, but fixed transformations (e.g., brightness, rotation) are easy for models to ignore and miss complex cases like agent interactions or road topology changes. With digital twin-supported generative models (e.g., Stable Diffusion), we can create diverse, realistic driving scenes, enabling richer transformations for more effective ADS testing under challenging conditions.

The validation of Autonomous Driving Systems (ADS) necessitates a comprehensive approach that accounts for both methodological rigor and real-world operational limitations. This section discusses three interrelated concepts critical to this validation process: the Operational Design Domain (ODD) [18], Metamorphic Testing (MT), and the application of generative models [19].

A. Operational Design Domain

ODD specifies conditions for ADS functionality, defined as a tuple: P (infrastructure), E (environment), O (constraints), T (temporal), and C (connectivity), as shown in Eq.1.

$$\text{ODD} = (P, E, O, T, C) \quad (1)$$

To capture the complexity of environmental factors, each component particularly the E (environment) can be broken down into the following parameters [20], as defined in Eq.2.

$$E = \{e_w \text{ (weather)}, e_l \text{ (lighting)}, e_v \text{ (visibility)}, e_t \text{ (temperature)}\} \quad (2)$$

B. Metamorphic Testing with ODD

Metamorphic Testing in a specified ODD involves the conditions that the metamorphic relations [21] should hold within certain operational limits. For any ADS S , input domain I , and output domain O , the ODD-constrained metamorphic relations are defined as in Eq.3.

$$\begin{aligned} \text{MR}_{\text{ODD}} \subseteq \{ & (x, S(x), x', S(x')) \mid \\ & x, x' \in I_{\text{ODD}}, \\ & R(x, S(x), x', S(x')) = \text{true} \} \end{aligned} \quad (3)$$

where I_{ODD} represents inputs valid within the ODD constraints, as defined in Eq.4.

$$I_{\text{ODD}} = \{x \in I \mid \forall c \in \text{ODD} : V(x, c) = \text{true}\} \quad (4)$$

Here, $V(x, c)$ verifies compliance with ODD constraint c .

C. Generative Models with ODD Integration

We extend the generative model framework to incorporate ODD constraints. For a generative model G and manually defined transformation specification τ , as shown in Eq. 5.

$$G(x, \tau, \text{ODD}) \rightarrow x' \text{ where } x, x' \in I_{\text{ODD}} \quad (5)$$

The transformation specification τ is now ODD-aware, as formally defined in Eq. 6:

$$\tau_{\text{ODD}} = \varepsilon \in E, \gamma \in P, \sigma \in O \times T \times C \quad (6)$$

This enables the definition of ODD-compliant metamorphic relations as shown in Eq. 7:

$$\begin{aligned} \text{MR}_{G, \text{ODD}} = \{ & (x, S(x), G(x, \tau_{\text{ODD}}), \\ & S(G(x, \tau_{\text{ODD}}))) \mid \\ & R(x, S(x), G(x, \tau_{\text{ODD}}), \\ & S(G(x, \tau_{\text{ODD}}))) = \text{true} \wedge \\ & x, G(x, \tau_{\text{ODD}}) \in I_{\text{ODD}} \} \end{aligned} \quad (7)$$

The formulation ensures synthesized cases remain within ODD limits, retain key ODD characteristics, and produce valid, meaningful results within the intended context.

Integrating ODD with MT and generative models to ensure that ADS testing remains relevant within the intended

conditions, while also upholding the rigor of mathematical validation [22].

III. PROPOSED APPROACH

ADSs demand thorough validation within their ODD. We propose a novel framework that integrates MT with generative AI to systematically validate ADS perception systems, addressing three key challenges: 1) Oracle problem [23] in ADS testing, 2) environmental complexity and scenario diversity, 3) uncertainty in perception systems.

Algorithm 1: ODD-Aware Digital Twin Scenario Generation

Input: Source image x , ODD specifications
Output: Transformed image x'
 Define τ based on ODD constraints;
 Verify transformation validity: $V(x, \tau) = \text{true}$;
 Generate candidate: $x' \leftarrow G_{\text{ODD}}(x, \tau)$;
if $\text{ValidateODDCompliance}(x')$ **then**
 | **return** x' ;
else
 | **return** $\text{GenerateNewTransform}(x)$;
end

A. Framework Overview

Our framework operates through a systematic workflow (Fig. 2) with two core components:

- **Digital Twin Scenario Generation:** Creates controlled variations in test scenarios while maintaining ODD compliance.
- **Metamorphic Validation:** Evaluates ADS behavior consistency under these variations.
- **Temporal Analysis:** Ensures consistency of predictions over time using smoothed outputs.

Formally, let S represent an ADS under test, as defined in Eq 8, where I denotes the input space (camera images), O represents the output space (driving decisions), and h, w , and c stand for image height, width, and channels, respectively:

$$S : I \rightarrow O, \text{ where } I \subset \mathbb{R}^{h \times w \times c} \quad (8)$$

The framework components are defined in Eq 9:

$$\begin{aligned} G_{\text{ODD}} : I \times \tau &\rightarrow I \\ V_{\text{MR}} : (I \times O) \times (I \times O) &\rightarrow \{0, 1\} \end{aligned} \quad (9)$$

where G_{ODD} generates ODD-compliant transformations and V_{MR} validates metamorphic relations.

B. ODD-Aware Digital Twin Scenario Generation

1) *Transformation Space:* For a source image $x \in I$, we define ODD-compliant transformations with the following components: Environmental conditions such as weather and lighting (ε), geometric transformations such as perspective

and scale (γ), and semantic modifications such as objects, road features (σ) as shown in Eq. 10.

$$\begin{aligned} \tau_{\text{ODD}} &= \{\varepsilon \in E, \gamma \in P, \sigma \in O\} \\ \text{subject to: } &\forall c \in \text{ODD} : V(G(x, \tau), c) = \text{true} \end{aligned} \quad (10)$$

2) *Generation Process:* involves generation of transformed images that adhere to ODD specifications, utilizing a visual generation model to produce metamorphic testing samples. This is detailed in Algorithm 1, and also illustrated as an integral component (1) of the metamorphic testing architecture for ADS in Fig. 2.

C. Metamorphic Relations and Validation

1) *Uncertainty-Aware Relations:* We enhance traditional MRs with uncertainty quantification as follows where $u(\cdot)$ denotes uncertainty quantification, θ_u represents the uncertainty threshold, and $R(\cdot, \cdot)$ stands for the relation validator as shown in Eq. 11:

$$\begin{aligned} \text{MR}_u(x, x') &= \{(S(x), S(x'), u(S(x)), u(S(x')))\} \mid \\ &R(S(x), S(x')) = \text{true} \wedge \\ &u(S(x')) \leq \theta_u \end{aligned} \quad (11)$$

2) *Validation Criteria:* For each MR category validation criteria are formulated as follows with the following three key components: Path extraction ($P(\cdot)$), Object detection ($D(\cdot)$), and tolerance thresholds (ϵ_p, ϵ_d).

MR1, MR2 and MR3 require that the error threshold is not exceeded, as the generator utilizes similar images for testing. The metamorphic relations are defined as Eq. 12

$$\begin{aligned} V_{\text{MR1},2,3}(x, x') &= \|P(S(x)) - P(S(x'))\| \leq \epsilon_p \\ V_{\text{MR1},2,3}(x, x') &= \|D(S(x)) - D(S(x'))\| \leq \epsilon_d \end{aligned} \quad (12)$$

- **MR1:** Maintain the same lane direction and angle, while slightly altering the background. The expectation is that the result of the test case should remain within an error margin of ϵ_p , since the key lane features are preserved.
- **MR2:** Keep the lane's direction and angle unchanged, but modify the weather conditions to snow. Snow partially obscures the road view, simulating a realistic environmental effect on perception algorithms. Despite the occlusion, the output should remain consistent with the original, within an acceptable error margin ϵ_d .
- **MR3:** Maintain the same lane direction and angle while narrowing the driving lane. The expected outcome of the test case should remain within an error margin of ϵ_e , as the primary direction of the lane features is preserved.

D. Temporal Analysis

The third key component of the metamorphic testing framework in Fig. 2 is time series analysis which aims to ensure robust validation across time sequences as formulated below where w , ϵ_t , and $S_t(\cdot)$ denote time window size, temporal

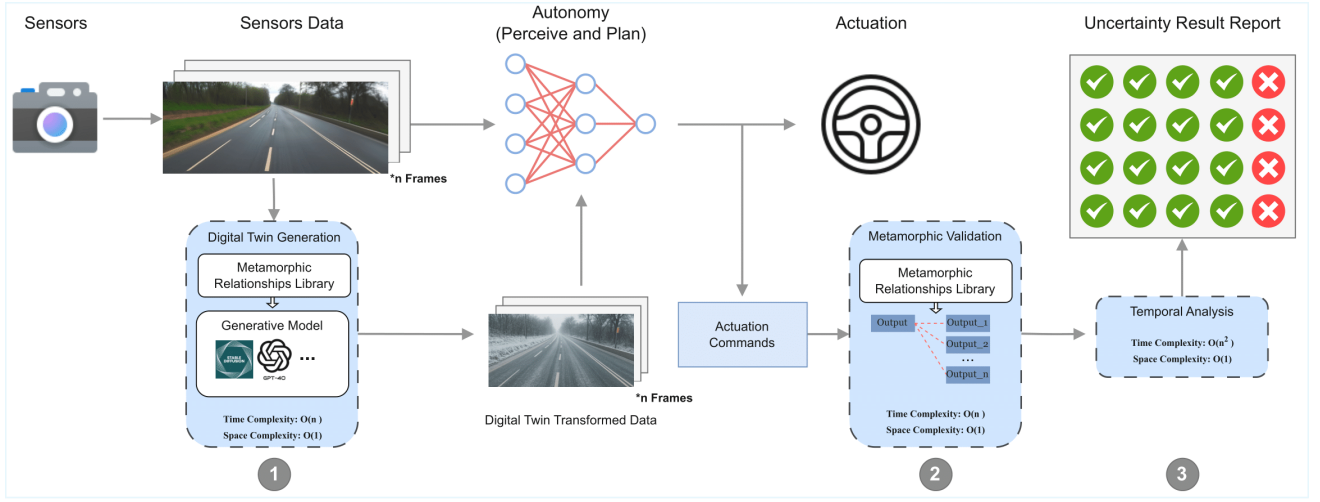


Fig. 2. Architecture of Digital Twin for metamorphic testing of ADS and its three key components: (1) Digital Twin Generation, (2) Integrated Validation, (3) Time Series Analysis.

threshold and smoothed prediction, respectively. The temporal validation condition is given in Eq 13.

$$S_t(x) = \frac{1}{w} \sum_{i=t-w}^t S(x_i) \quad (13)$$

$$V_{\text{temporal}}(x, x') = \|S_t(x) - S_t(x')\| \leq \epsilon_t$$

E. Integrated Validation Framework

Integrated validation evaluates if outputs meet safety standards by utilizing metamorphic relations, as mathematically defined and described in Algorithm 2. This framework acts as the second essential part of the metamorphic testing architecture for ADS, as depicted in Fig. 2

Algorithm 2: Integrated Validation Framework

Input: Image sequence X , ADS S , ODD specifications

Output: Validation report R

Initialize empty report R ;

foreach $x_t \in X$ **do**

 Generate ODD-compliant τ_t ;

$x'_t \leftarrow G_{\text{ODD}}(x_t, \tau_t)$;

$s_t \leftarrow S(x_t)$;

$s'_t \leftarrow S(x'_t)$;

$u_t \leftarrow \text{ComputeUncertainty}(s_t)$;

$u'_t \leftarrow \text{ComputeUncertainty}(s'_t)$;

$v_{\text{mr}} \leftarrow \text{ValidateRelations}(s_t, s'_t)$;

$v_{\text{temp}} \leftarrow \text{TemporalValidation}(s_{t-w:t}, s'_{t-w:t})$;

 UpdateReport($R, v_{\text{mr}}, v_{\text{temp}}, u_t, u'_t$);

end

return R ;

This systematic framework enables rigorous validation of ADS perception systems while maintaining practical relevance within specified operational bounds.

IV. EMPIRICAL EVALUATION

A. Procedure

We generated a dataset of transformed driving scenarios using Stable Diffusion-XL as described in Section III. For each base driving frames within the dataset, we applied controlled variations to simulate diverse ODD conditions such as fog, rain, glare, and darkness. Each ODD-compliant transformation was validated using Algorithm 1, ensuring operational integrity.

B. Setups

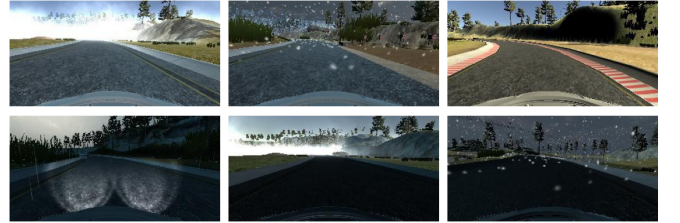


Fig. 3. Sample scenes from test dataset, which includes diverse driving scenarios across different times of day and weather conditions.

To unify the measurement standards, we use the dataset that contains crash cases collected in the Udacity simulator under various weather and conditions, which is the same as that used in the Self Oracle project [15]. These episodes encompass a wide range of environmental conditions and temporal settings, including variations such as Daynight, Daynightfog, Daynight-rain, and Daynightsnow, as well as isolated conditions like Fog, Rain, Snow, and Normal weather. This comprehensive coverage enables robust evaluation of the model's performance under realistic and challenging circumstances.

For our test model, we adopted the DAVE-2 architecture with nine layers: one normalization, five convolutional, and three fully connected layers. Input images were split into YUV

planes and the model was trained on the Udacity Simulator Dataset across three tracks with dropout rates between 0.1 and 0.5. For metamorphic test generation, we used Stable Diffusion XL with 16-bit precision, processing 1024×512 pixel images with transformation strength 0.2, guidance scale 10.0, and negative prompts excluding “low quality, distorted, cartoonish, unrealistic” content.

C. Results

Method	TP	FP	TN	FN	TPR	FPR	F1	Prec.
<i>SelfOracle</i>								
CAE	0	0	4282	196	0	0	n.a.	n.a.
DAE	21	55	4063	175	0.107	0.013	0.154	0.276
SAE	108	183	2665	88	0.551	0.064	0.444	0.371
VAE	107	183	2959	89	0.546	0.058	0.440	0.369
LSTM	7	12	4119	186	0.036	0.003	0.066	0.368
DeepRoad	44	250	3651	152	0.225	0.064	0.180	0.150
<i>Stable Diffusion</i>								
MR1	137	77	403	59	0.699	0.160	0.668	0.640
MR2	141	72	408	55	0.719	0.150	0.689	0.662
MR3	135	87	393	61	0.689	0.181	0.645	0.608

TABLE I
EVALUATION RESULTS FOR ALL VARIANTS.

Compared to Self-Oracle and DeepRoad, our Stable Diffusion variants (MR1,MR2,MR3) show notable gains in TPR, F1, and Precision—key metrics for reliable, accurate prediction in safety-critical applications.

Among the evaluated methods: MR2 consistently outperforms all other strategies, achieving the highest TPR (0.719), F1 score (0.689), and Precision (0.662). This suggests that MR2 is not only more accurate in detecting true crash scenarios but also less prone to false alarms compared to other methods.

MR1 and MR3 also show robust performance, with F1 scores of 0.668 and 0.645, respectively. While MR3 shows a slightly higher TPR than MR1 (0.689 vs. 0.699), its FPR is also marginally higher, indicating a small trade off between sensitivity and specificity. In contrast, traditional Self-Oracle approaches such as VAE, DAE, SAE, and DeepRoad fall behind. For instance, DeepRoad has a TPR of only 0.225 and an F1 score of 0.180, reflecting weaker detection capabilities. These improvements are further illustrated by Figure 4, which tracks the number of successful crash predictions over time for each MR variant. The plot illustrates that:

MR3 highlights the most favorable early crash prediction performance, with a distribution that is more concentrated in earlier frames, particularly within the critical 5-second pre-crash window (represented by the dashed blue line). Although MR2 achieves the highest total number of successful predictions (141 out of 196), its predictions are more concentrated closer to the crash event, similar to MR1. MR1 and MR2 exhibit comparable patterns in both timing and peak predictive success, though they are slightly less proactive than MR3 in anticipating crashes.

The time series trend highlights the proposed method’s ability not only to respond to imminent crashes but also to act

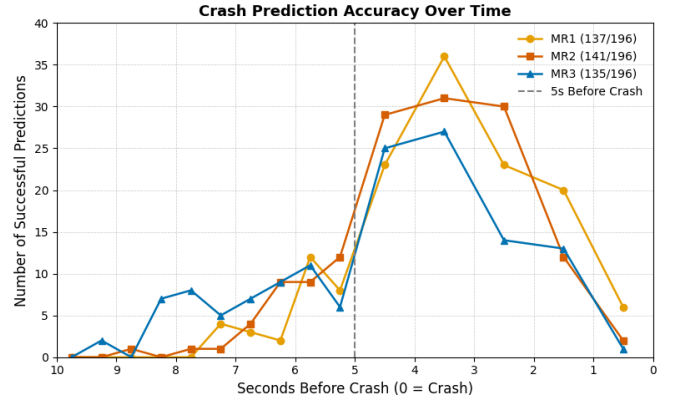


Fig. 4. Distribution of successful crash predictions made by each MR variant across time leading up to the crash.

as a proactive safety mechanism, detecting potential hazards well before they occur.

These findings demonstrate the method’s superior effectiveness, strong early warning capability, and consistent reliability compared to traditional approaches. They also confirm its suitability for real-world deployment in autonomous systems and other safety-critical settings, where timely and accurate hazard prediction is vital.

D. Extensions and Challenges of the Proposed Framework

By integrating a generative model, our framework enables the design of flexible and adaptive metamorphic relations beyond the three initially proposed. Due to the use of the generative model and the digital twin feature, our framework offers high flexibility in creating a variety of metamorphic rules. Owing to content limitations, we explored the three most representative and practical metamorphic relations in this paper. As shown in Table II, we list the potential for the proposed framework to generate other efficient test cases.

Although the framework shows exceptional performance, the significant processing power demands of Stable Diffusion models create computational hurdles for real-time implementation. Consequently, the approach is presently best suited for closed-loop testing settings during the development and certification stages of ADS. With advancements in generative model technology and improvements in computational efficiency via methods like model compression and specialized hardware acceleration, the framework’s use will broaden to include real-time tracking and production implementation in autonomous vehicles

V. CONCLUSION

This paper presents a Digital Twin based approach to advancing the validation of autonomous driving systems (ADS) by enabling systematic evaluation of system behavior across a wide range of driving scenarios, including rare and safety-critical edge cases. By creating a virtual replica of the ADS and its operational environment, the framework facilitates controlled, repeatable testing through reusable metamorphic

TABLE II
POTENTIAL METAMORPHIC RELATIONS FOR AUTONOMOUS DRIVING SCENARIOS

MR	Name	Transformation (τ)	Description
MR4	Agent Substitution	$\tau_{\text{agent}}(x) = \text{GenerateAgent}(x, \text{new_type}, \text{preserve}=(\text{position}, \text{velocity}, \text{size_class}))$	Replace traffic participants with similar-sized agents (car $\leftrightarrow$ truck, bike $\leftrightarrow$ motorcycle) while preserving position and velocity
MR5	Time-of-Day Consistency	$\tau_{\text{time}}(x) = \text{AdjustLighting}(x, \text{target_time}, \text{preserve}=(\text{geometry}, \text{objects}, \text{lanes}))$	Transform day scenes to night/dusk while preserving all geometric and semantic content
MR6	Traffic Control Equivalence	$\tau_{\text{control}}(x) = \text{ReplaceSignal}(x, \text{equivalent_control}, \text{preserve}=(\text{intersection_geometry}))$	Replace equivalent traffic control devices (stop sign $\leftrightarrow$ red light)
MR7	Emergency Vehicle Priority	$\tau_{\text{emergency}}(x) = \text{ReplaceEmergency}(x, \text{new_type}, \text{preserve}=(\text{priority}, \text{position}, \text{signals}))$	Replace emergency vehicle type while maintaining priority status and position
MR8	Construction Zone Adaptation	$\tau_{\text{construction}}(x) = \text{AddConstruction}(x, \text{cone_pattern}, \text{preserve}=(\text{intended_path}, \text{lane_width}))$	Transform normal lanes to construction zones with equivalent path guidance
MR10	Obstacle Substitution	$\tau_{\text{obstacle}}(x) = \text{ReplaceObstacle}(x, \text{equivalent_obstacle}, \text{preserve}=(\text{size}, \text{position}, \text{blockage}))$	Replace static obstacles with equivalent objects (fallen tree $\rightarrow$ barrier)

relations and robust evaluation metrics. This digital twin integration enhances safety assurance and supports the development of resilient machine learning components for real-world deployment. Through experiments conducted in the Udacity Simulator, we implemented and evaluated the framework against baseline methods such as Self-Oracle and DeepRoad. Among the proposed metamorphic relations, MR2 yielded the highest true positive rate (0.719), F1 score (0.689), and precision (0.662), while MR3 exhibited strong early crash prediction performance, with accurate detections concentrated before the critical 5-second mark. These findings highlight the effectiveness of our digital twin-driven method in identifying hazardous behaviors with improved accuracy and responsiveness. Overall, we demonstrate that incorporating digital twins into metamorphic testing provides a scalable and high-fidelity strategy for safety verification in autonomous vehicles. As ADS technologies evolve, the framework's virtual mirroring capability is expected to enable even more proactive, efficient, and context-aware safety assessments in future mobility.

ACKNOWLEDGEMENT

This work is supported in part by MITACS Accelerate Program under project IT40981, in part by the NSERC CREATE TRAVERSAL program, and in part by Ontario Research Fund-Research Excellence under grant number RE 12-026.

REFERENCES

- [1] J. Zhao, W. Zhao, B. Deng, Z. Wang, F. Zhang, W. Zheng, W. Cao, J. Nan, Y. Lian, and A. F. Burke, "Autonomous driving system: A comprehensive survey," *Expert Systems with Appl.*, p. 122836, 2023.
- [2] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2021.
- [3] T. Y. Chen, F.-C. Kuo, H. Liu, P.-L. Poon, D. Towey, T. Tse, and Z. Q. Zhou, "Metamorphic testing: A review of challenges and opportunities," *ACM Computing Surveys (CSUR)*, vol. 51, no. 1, pp. 1–27, 2018.
- [4] S. Mihai, M. Yaqoob, D. V. Hung, W. Davis, P. Towakel, M. Raza, M. Karamanoglu, B. Barn, D. Shetve, R. V. Prasad, *et al.*, "Digital twins: A survey on enabling technologies, challenges, trends and future prospects," *IEEE COMST*, vol. 24, no. 4, pp. 2255–2291, 2022.
- [5] T. Zhang, B. Kantarci, and U. Siddique, "Augmented metamorphic testing for comprehensive validation," in *Proc. Workshop on Software Engineering for Autonomous Driving Systems (SE4ADS) at the 47th IEEE/ACM Intl Conf on Software Engineering*, Apr. 2025.
- [6] A. Niaz, M. U. Shoukat, Y. Jia, S. Khan, F. Niaz, and M. U. Raza, "Autonomous driving test method based on digital twin: A survey," in *2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, pp. 1–7, 2021.
- [7] S. Segura, G. Fraser, A. B. Sanchez, and A. Ruiz-Cortés, "A survey on metamorphic testing," *IEEE Transactions on software engineering*, vol. 42, no. 9, pp. 805–824, 2016.
- [8] X. Gao, Z. Wang, Y. Feng, L. Ma, Z. Chen, and B. Xu, "Multitest: Physical-aware object insertion for testing multi-sensor fusion perception systems," in *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, ICSE '24, (New York, NY, USA), Association for Computing Machinery, 2024.
- [9] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," 2021.
- [10] H. Cao, C. Tan, Z. Gao, Y. Xu, G. Chen, P.-A. Heng, and S. Z. Li, "A survey on generative diffusion models," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 7, pp. 2814–2830, 2024.
- [11] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019.
- [12] L. Kumar and A. Kumar, "Securing digital twins in autonomous vehicles: Cyber threats, mitigation strategies, and innovative security solutions," in *2025 Second International Conference on Cognitive Robotics and Intelligent Systems (ICC - ROBINS)*, pp. 146–151, 2025.
- [13] T. Samak, C. Samak, S. Kandhasamy, V. Krovi, and M. Xie, "Autodrive: A comprehensive, flexible and integrated digital twin ecosystem for autonomous driving research & education," *Robotics*, vol. 12, no. 3, p. 77, 2023.
- [14] J. Wu, Z. Huang, P. Hang, C. Huang, N. De Boer, and C. Lv, "Digital twin-enabled reinforcement learning for end-to-end autonomous driving," in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, pp. 62–65, 2021.
- [15] A. Stocco, M. Weiss, M. Calzana, and P. Tonella, "Misbehaviour prediction for autonomous driving systems," in *Proc. ACM/IEEE 42nd Intl Conf on Software Engineering*, pp. 359–371, ACM, 2020.
- [16] J. Wang, Y. Ishimoto, M. Kondo, Y. Kamei, and N. Ubayashi, "Uncertainty-aware metamorphic testing for robust object detection models," in *2023 IEEE 23rd International Conference on Software Quality, Reliability, and Security Companion (QRS-C)*, IEEE, 2023.
- [17] J. Ayerdi, A. Iriarte, P. Valle, I. Roman, M. Illarramendi, and A. Arrieta, "Marmot: Metamorphic runtime monitoring of autonomous driving systems," *ACM Transactions on Software Engineering and Methodology*, vol. 34, no. 1, pp. 1–35, 2024.
- [18] "Road vehicles — test scenarios for automated driving systems — specification for operational design domain," 2023.
- [19] A. Oussidi and A. Elhassouny, "Deep generative models: Survey," in *International conference on intelligent systems and computer vision (ICSV)*, pp. 1–8, IEEE, 2018.
- [20] L. Mendiboure, M. L. Benzagouta, D. Gruyer, T. Sylla, M. Adedjouma, and A. Hedhli, "Operational design domain for automated driving systems: Taxonomy definition and application," in *2023 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1–6, 2023.

- [21] A. Lemesle, A. Varasse, Z. Chihani, and D. Tachet, *AIMOS: Metamorphic Testing of AI - An Industrial Application*, pp. 328–340. 09 2023.
- [22] Z. Jiang, J. Liu, P. Sun, M. Sang, H. Li, and Y. Pan, “Generation of risky scenarios for testing automated driving visual perception based on causal analysis,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 15991–16004, 2024.
- [23] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, “The oracle problem in software testing: A survey,” *IEEE transactions on software engineering*, vol. 41, no. 5, pp. 507–525, 2014.