

南京大學

研 究 生 毕 业 论 文 (申请硕士学位)

 论 文 题 目
 基于特定领域编程语言的自动驾驶交通场景数据生成与模型测试技术

 作者姓名
 夏志龙

 专业方向
 软件工程

 研究方向
 工学硕士(软件工程领域)

 指导教师
 陈振宇教授,冯洋助理研究员

2022年05月20日

学 号 : MG1932014
论文答辩日期 : 2022年05月20日
指 导 教 师 : (签字)



Traffic scene data generation and model testing technology for autonomous driving based on domain-specific programming language

By

Zhilong Xia

Supervised by

Professor Zhenyu Chen Assistant Professor Yang Feng

A Thesis Submitted to the Software Institute and the Graduate School of Nanjing University in Partial Fulfillment of the Requirements for the Degree of **Master**

> Software Institute May 2022

学位论文原创性声明

任何收存和保管本论文的单位和个人,未经作者本人授权,不得将本论文 转借他人并复印、抄录、拍照或以任何方式传播,否则,引起有碍作者著作权益 的问题,将可能承担法律责任。

本人郑重声明:所呈交的学位论文,是本人在导师的指导下,独立进行研究 工作所取得的成果。除文中已经注明引用的内容外,本论文不句含其他个人或 集体已经发表或撰写的作品成果。本文所引用的重要文献,均已在文中以明确 方式标明。本声明的法律结果由本人承担。

论文作者签名:_____

日期: 年 月 日

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目:基于特定领域编程语言的自动驾驶交通场景数据生成与模型测试技术

软件工程	专业_2019_级硕士生姓名:
指导教师 (姓名、职称):	陈振宇 教授, 冯洋 助理研究员

摘要

自动驾驶交通感知相关模型的可靠性与安全性等质量问题已经变得越来越 重要。感知模型系统的测试验证依赖大规模的场景样本,然而,目标交通领域的 数据采集受到成本、时间效率和环境限制的影响,难以获得大规模、高质量的测 试数据。同时,现有的一些数据增强方法存在一定的局限性,例如数据硬编码变 换的合理性问题,模拟数据标注问题。另一些研究基于仿真工具进行系统测试 验证,但是具体感知场景的构建与生成缺乏有效的引导策略。

本文提出一种基于特定领域编程语言 Scenic 的自动驾驶交通场景数据生成 与模型测试技术。该技术对于道路场景构建代码中交通实体、状态以及属性进 行变异,变异 Scenic 代码用于引导仿真模拟器生成具体交通场景测试数据,一 定程度上避免了一些数据硬编码扩增方法带来的潜在的可解释性问题,同时能 够大规模生成真实有效的测试数据。研究创建了用于变异的 Scenic 种子场景库, 分析了 Scenic 场景构建代码中可变异内容与变异模式。提出了针对两种典型自 动驾驶感知相关模型的测试验证方法,针对自动驾驶道路目标识别模型,设计 基于规则的变异测试方法,使用特定脚本变异规则引导代码变异,将规则中的 变异前后图像约束关系用于验证测试结果。针对自动驾驶行为预测模型,搭建 基于 VerifAI 的测试验证模块,将待测模型,变异代码生成的动态测试数据以 及评估验证指标进行组合,以发现模型预测结果中存在的偏差。通过具体实验 验证了数据生成和模型测试技术的有效性,实验成功发现了道路目标识别模型 Yolo-v4 中目标识别漏检、误检、以及置信度过低问题。证明了基于规则的变异 测试能够不依赖数据标注, 高效地发现模型不一致输出, 并用于进行模型的数 据增强。实验还验证了基于 VerifAI 搭建的自动驾驶行为预测模型测试模块,对 于 LaneGCN 模型进行变异场景测试,多种性能指标计算表明,本技术构造的变 异扩增样本拥有较高的质量,能够发现模型潜在的缺陷。

关键词:自动驾驶,自动驾驶测试,图像生成,领域特定语言,仿真

i

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Traffic scene data generation and model testing technology for autonomous driving based on domain-specific programming language

SPECIALIZATION: Software Engineering

POSTGRADUATE: Zhilong Xia

MENTOR: Professor Zhenyu Chen Assistant Professor Yang Feng

Abstract

Quality issues such as reliability and safety of models related to autonomous driving traffic perception have become increasingly important. The test validation of perception model systems relies on large-scale scene samples. However, data collection in the target traffic field is affected by cost, time efficiency, and environmental constraints, making it difficult to obtain large-scale, high-quality test data. At the same time, some existing data enhancement methods have certain limitations, such as the rationality of hard-coded transformation of data and the problem of simulating data labeling. Other studies are based on simulation tools for system testing and verification, but the construction and generation of specific perception scenes lacks effective guidance strategies.

This paper proposes a data generation and model testing technology for autonomous driving traffic scenarios based on the domain-specific programming language Scenic. This technology mutates the traffic entities, states and attributes in the road scene construction code, and the mutated scene code is used to guide the simulator to generate specific traffic scene test data. To a certain extent, the potential interpretability problems caused by some data hard-coded amplification methods are avoided, and at the same time, real and effective test data can be generated on a large scale. The research creates a Scenic seed scene library for mutation, and analyzes the variable content and mutation pattern in the code. A test and verification method for two typical autonomous driving perception related models is proposed. For the autonomous driving road target

recognition model, a rule-based mutation testing method is designed, which uses specific script mutation rules to guide code mutation, and uses the constraints before and after mutation in the rules to verify the test results. For the autonomous driving behavior prediction model, a test verification module based on VerifAI is built, which combines the model to be tested, the dynamic test data generated by the mutated code, and the evaluation and verification indicators to discover the deviations in the model prediction results. The effectiveness of the data generation and model testing technology is verified through specific experiments. The experiments successfully found the problems of missed detection, false detection, and low confidence in the road target recognition model Yolo-v4. It is proved that the rule-based mutation test can efficiently find the inconsistent output of the model without relying on data annotation, and can be used for data enhancement of the model. The experiment also verified the test module of the automatic driving behavior prediction model built based on VerifAI. The LaneGCN model is tested in variant scenarios and the calculation of various performance indicators shows that the variant amplification samples constructed by this technology have high quality and can find potential defects of the model.

Keywords: autonomous driving, autonomous driving testing, image generation, domain specific language, simulation

J.

表	目表	录 · · · · · · · · · · · · · · · · · · ·		vii
冬	日素	录	····· v	iii
第-	一章	绪论		1
	1.1	选题的背景和意义		1
	1.2	国内外研究现状及分析		2
	1.3	本文工作		4
	1.4	本文的组织结构		5
第_	二章	相关背景技术概述		6
	2.1	自动驾驶系统		6
		2.1.1 道路目标检测		8
		2.1.2 自动驾驶行为预测 ·		9
	2.2	自动驾驶测试		10
		2.2.1 仿真测试环境建模 ·		11
		2.2.2 数据驱动测试 ·····		13
		2.2.3 传感器模拟		15
		2.2.4 领域编程语言与自动	驾驶测试	17
	2.3	本章小结		18
第三	三章	基于 Scenic 的自动驾驶交通	场景数据生成与模型测试技术	20
	3.1	Scenic 场景代码构建方法 ··	••••••	21
		3.1.1 Scenic 预处理与代码	规范化	21
		3.1.2 Scenic 脚本建模 ····	:	24
		3.1.3 典型场景构建	••••••	28
		3.1.4 脚本解析与变异对象	分析	28
	3.2	基于变异规则的道路目标识	别模型测试方法	33

	3.2.1	变异规则引导的样本生成	34
	3.2.2	模型测试与增强方法	34
3.3	基于 V	erifAI 的车辆行为预测模型测试方法	38
	3.3.1	仿真数据实时采集	38
	3.3.2	车辆轨迹预测模型	39
	3.3.3	VerifAI下 Scenic 变异扩增脚本测试方法	40
第四章	实验验	证与分析	45
4.1	实验准	备	45
	4.1.1	数据筛选与采样	45
	4.1.2	数据预处理	47
4.2	实验1	: 基于变异的道路目标识别模型缺陷检测	49
	4.2.1	数据说明	49
	4.2.2	实验设计	52
	4.2.3	结果分析	53
4.3	实验 2	: 基于规则的道路目标识别模型测试与增强	55
	4.3.1	数据准备	55
	4.3.2	实验设计 ·····	56
	4.3.3	结果分析	56
4.4	实验3	: 基于变异的自动驾驶行为预测模型验证	60
	4.4.1	数据准备	61
	4.4.2	实验设计 ·····	62
	4.4.3	实验3结果分析 ······	64
4.5	有效性	分析和讨论	65
4.6	分析总	结	66
第五章	总结和	展望	68
5.1	总结…		68
5.2	工作展	望	68
参考文献	献		70
简历与和	斗研成界	Ļ	81

目录	vi
致谢	82

表目录

3.1	使用的 Scenic 脚本模板	23
3.2	使用的描述符和算子语法及其含义	26
3.3	包含基本语法的种子场景	29
3.4	变异内容与变异方法	32
3.5	Scenic 样本变异规则设计	35
3.6	用于评估变异映射关系的方法	36
3.7	样本分类图	37
4.1	实验软硬件环境配置	46
4.2	Udacity 数据集类别说明 ······	50
4.3	变异结果	54
4.4	8 种变异规则下测试实验结果	58
4.5	Yolo 模型性能:AP 指标 ······	59
4.6	Yolo 模型性能:AR 指标	59
4.7	VerifAI 验证 LaneGCN 的各指标计算结果	64

图目录

1.1	特斯拉和 Uber 的自动驾驶汽车均发生过严重事故	2
2.1	模块化自动驾驶系统架构	7
3.1	总体技术架构图	20
3.2	同一份 Scenic 代码生成的多种场景	22
3.3	使用原始 Scenic 渲染得到的一个场景,该 Scenic 代码仅使用了 1	25
	行代码进行场意定义	25
3.4	Scenic 脚本语法解析树分析和可视化 ······	31
3.5	执行变异后不同变异脚本仿真渲染结果中采样的近似场景图片	32
3.6	基于变异规则的道路目标识别模型测试方法示意图	33
3.7	目标检测中 IoU 与检测框示例	36
3.8	基于 VerifAI 的模型测试方法示意图	39
3.9	LaneGCN[1] 网络结构 · · · · · · · · · · · · · · · · · · ·	41
4.1	图像白化处理	49
4.2	随机变异下误检问题	52
4.3	随机变异下漏检问题 · · · · · · · · · · · · · · · · · · ·	53
4.4	预训练模型性能变化	57
4.5	使用非一致性增强数据集后训练模型性能变化	60
4.6	Argoverse 行为预测数据集中的一个动态场景	61
4.7	使用 seed6.scenic 变异生成的 3 种动态场景	62
4.8	模型预测示意图	63

第一章 绪论

1.1 选题的背景和意义

近几年来,随着人工智能、网络通信和车载传感器(尤其是激光雷达)等 关键技术的进步和突破,智能驾驶 [2] 和自动驾驶系统 [3] 已经越来越受到国内 外汽车厂商和研究机构的重视。高级别的自动驾驶系统通过传感器采集的环境 数据获取车辆自身信息和周围环境信息并进行数据融合,目标识别模型与行为 预测模型 [4] 则对环境信息进行分析,实现道路目标识别检测和车辆行为轨迹预 测,从而进一步帮助决策控制系统作出一系列的命令。可以看出,理解环境数据 是自动驾驶实现的基础,自动驾驶算法模型往往基于数据驱动,其训练和测试 验证需要大量的高质量交通场景数据,一旦数据本身存在质量问题,必然影响 相关模型的可靠性。必须重视这类自动驾驶模型本身的可靠性 [5],其潜在缺陷 将会产生包含人员生命和财产安全损失的严重后果 1.1。2018 年 Uber 自动驾驶 汽车在美国进行路测时,与一名横穿马路的女性相撞并致其抢救无效死亡。当 时该女性正骑行自行车横穿马路,正是由于感知系统未能在夜晚正确识别目标 对象,并做出防止碰撞的决策,才最终导致了这一惨剧。由此可见,感知算法模 型质量保障的重要性不容忽视。然而,现实生活中,传统的测试方法[6]往往很 难对基于深度学习的 AI 软件模型进行测试,因为其模型代码与功能实现并非如 传统软件一样存在耦合关系,很难凭借一个神经网络模型的层级结构和判断其 功能的"好"与"差"。同时, AI 模型运行机理不透明, 部分细节缺乏可解释性, 模型搭建和训练过程甚至被称为"炼丹",就可以预见其测试的难度。

当前较为有效的自动驾驶测试是数据驱动测试 [7],即利用符合真实场景特征的大规模高质量(满足数据多样性、分布合理性)测试数据进行模型测试验证,这项工作可以在模型训练测试阶段开展,也可以由其他专业测试人员进行。但是,这种方法最大问题来自测试数据本身,通过真实道路驾驶采集感知模型训练测试数据样本或进行数据标注方法往往受到驾驶成本和真实环境条件的客观因素限制,在数据规模和数据质量上难以满足模型需求。早在 2010 年, Campbell等人 [8]就提出了数据是影响自动驾驶的重要因素。为了解决真实道路数据不足的问题,业内的另一大趋势是使用虚拟仿真数据 [9],这带来了新的问题,如何利用虚拟仿真构造有效数据?如何保障通过虚拟方式构造的感知数据质量?以及缺乏真实交通环境 Oracle 情形下,如何进行有效的测试与验证?本文研究课题尝试解决这些问题,该课题围绕利用图像感知领域概率编程语言 [10] 进行交



图 1.1: 特斯拉和 Uber 的自动驾驶汽车均发生过严重事故

通场景建模,利用约束和仿真模拟器规则限制保障生成虚拟样本的有效性,并 设计了用生成样本进行感知[11]和行为预测模型[1]的验证模块。通过本课题研 究,尝试扩展自动驾驶虚拟测试新样本构建思路,探究虚拟仿真测试能否有效 发现已有自动驾驶相关模型潜在缺陷问题。

1.2 国内外研究现状及分析

基于 AI 的自动驾驶模型性能不断提升,当前越来越多的汽车厂家、互联网 企业自动驾驶研究部门推出了各自的自动、驾驶解决方案,在较低的自动驾驶 级别(L3 及以下)中取得了良好的效果,学术界和工业界也在不断尝试创建支 持自动驾驶级别更高的技术与模型。这些技术和模型的安全性、鲁棒性等质量 影响自动驾驶的发展和产品落地,因此,针对自动驾驶关键模型的质量保障研究 [12] 也成为了当下的软件工程研究者的热门研究方向,其中对于感知模型、以 及基于感知的道路目标识别与驾驶轨迹分析预测模型的测试验证方法不断被提 出。这些模型依赖来自感知相关组件的有效数据,包括来自车载传感器的图像、 雷达数据和地理信息系统数据,从而构建完整的自动驾驶场景。然而,由于真实 交通场景数据获取难度大,成本较高,且受到数据采集技术和采集场景客观环 境多样性不足等各种因素限制,使用真实场景数据进行感知和相关分析预测模 型的验证是不充分的[13],使用仿真模拟数据是一种重要的解决思想,尤其是针 对数据驱动的深度神经网络模型。

很多的研究关注有效的测试度量指标和对应测试数据生成模式,包括神经 元覆盖率[14,15]、神经元边界[16,17],以及相关神经网络结构化覆盖指标的有 效性问题 [18, 19]。这些测试度量指标有效地配合了不同的测试验证技术: Deep-Xplore[14] 提出基于多模型差分测试的验证方案,将搜索不一致输出和最大化 神经元覆盖指标进行联合优化,以检测分析车道图像的转向模型的潜在风险。 DeepTest[20] 则重视测试数据在流程中的作用,利用多种数据变换生成探索神经 网络决策边界的测试用例。Zhang[21] 则使用蜕变测试思想引导交通场景图像合 成和验证。还有很多研究探索特定交通领域下测试用例的启发式搜素生成,相 关研究包括 Li 等人 [17] 进行的不利环境风格迁移下的极端环境场景搜索生成技术,Calo 等人 [22] 进行的基于搜索的碰撞避免配置检测技术。

使用能够进行完整的模拟流程的虚拟环境进行自动驾驶测试被广泛采用, 已经出现了多种功能丰富的自动驾驶仿真引擎,如 AirSim[23]、Carla[24],支持 自动驾驶静态环境、动态交通行为和交通实体的物理仿真,这些仿真器极大地 推动了自动驾驶虚拟测试框架和工具开发 [25-27]。研究人员提出了多种交通环 境建模方法,[9,28,29]等的研究中都使用来自真实场景的数据进行场景建模, 构建多样化的虚拟测试场景。Li[30]和 Musat[31]的工作则关注自动驾驶场景的 合成与变异,以提供更加细致的自动驾驶测试样本。传感器仿真是一类重要的 测试数据生成方法,研究者们提出了针对 RGB 摄像头数据 [32-34]、雷达点云 数据 [35,36]、轨迹数据 [37]和温控电路数据 [38]的仿真测试方法。Wong 等 人 [39]提出的方法使用仿真替代传感器模拟,直接模拟自动驾驶车辆的感知和 预测系统的输出数据来进行仿真测试,在训练数据上使用真实感知预测输出和 groundtruth 成对构建。Yang 等人 [40]使用有限的初始车辆的雷达和摄像头数据 进行数据变异,该方法首先基于纹理映射面元将这些数据中重要信息进行提取 (交通参与对象三维位置信息、场景特征信息),利用对抗生成网络重建新的测 试场景。

自动驾驶领域特定编程语言的出现提高了测试场景构建的效率和质量。领 域特定语言能够支持使用简单的语法规则设计目标交通场景,将复杂的专家知 识和领域客观真理转化为代码约束,并具有良好的可扩展性。国内外研究中, 基于领域特定语言的测试数据构建近年来进展快速。Fremont[10]等人开发的 Scenic语言简化了编写复杂交通场景的任务,并支持使用专门的采样技术,以生 成有效的交通场景。该语言支持定义场景中的环境配置、空间对象配置、分布、 交通参与者具体属性和非线性相对位置关系,还允许用户以声明的方式为具体 的场景构建添加对应约束。Paracosm[41]支持用户以编码方式描述具有特定特 征的复杂驾驶场景,个性化调整不同的参数配置,如道路布局、场景环境条件、 交通参与对象的反馈和响应。作者使用基于组合测试的方法,对于不同的参数 配置进行了模糊测试,同时还定义了对应的测试覆盖概念。O'Kelly[42] 则提出 了一种场景描述语言来创建具体的驾驶场景,支持定义交通场景中的不同代理 对象以及实际的道路拓扑结构,形式化地定义时间逻辑下的正确性约束,设计 的自动化测试数据生成工具处理可执行场景代码,对目标场景进行创建,可以 发现自动驾驶系统违反规范的行为,并提高测试样本覆盖率。

国内外研究表明,当前自动驾驶系统质量保障研究已经变得越来越重要,不 同测试技术和概念提出推动了该领域的发展。其中不同的技术方法存在其优缺 点,数据驱动测试方法十分有效,但是测试数据构建需要考虑使用可信指标进 行数据质量评估,同时大规模生成测试数据时模型测试结果的验证分析是一项 重要挑战。交通场景数据的变异方法则极大地增加了可以用于测试的样本,并 在数据的多样性增强和极端场景攻击样本生成任务上具有良好的效果,但是直 接针对数据的变换,如硬编码数学变换和对抗生成式变换后数据质量和有效性 存在一定的争议。虚拟仿真测试减少了自动驾驶测试中人工成本和质量问题,适 用于整车和感知组件的测试,使用领域特定语言则进一步提高了软件工程研究 者和自动驾驶专家参与自动驾驶质量保障的效率,但是仿真场景的设计依赖有 效的设计模式和规则。

1.3 本文工作

本论文提出了一种基于特定领域编程语言 [10] 的自动驾驶交通场景数据生成与模型测试技术,该技术借助代码分析与变异模式设计,变异用于渲染交通 图像的初始 Scenic 脚本代码,获得大量构建变异场景脚本代码,基于规则分析 和评估不同类型的变异样本,实现了对多种变异模式的测试样本的有效性验证, 并将其应用于测试和增强目标自动驾驶目标识别模型,还借助模拟器引擎与 AI 验证工具进一步设计了面向复杂自动驾驶行为预测模型的测试验证方法。

总体来讲,本文的主要工作和贡献如下:

- 基于特定领域编程语言 Scenic,通过对于调用渲染引擎的场景构建语言直接进行场景变异,相对在图像上进行直接变换的变异生成方法更加合理,尽量避免可解释可验证性问题,针对特定交通场景对象的变异也更加直接丰富。
- 将变异规则与真实的交通环境规则进行对应,提供典型交通场景的变异方案,更加具有合理性,贴近真实场景,详细的变异规则提供了有效的大规 模测试数据生成和测试验证的方法引导。
- 3. 基于渲染后的变异样本,设计了完善的自动驾驶感知相关模型测试验证方

1.4 本文的组织结构

案,对自动驾驶相关图像感知和基于感知的行为预测模型进行充分验证, 并进行了模型增强的实验。

1.4 本文的组织结构

本论文的组织形式如下:

- 第二章 分析和解释本研究涉及的背景知识和相关技术原理,对于该目标或研究 领域的学术理论研究方法和工业实践的发展历程和重要成果,以及存在的 挑战,最后归纳和总结了相关背景知识,并描述了本文研究内容与相关背 景知识的关联和提出方法的价值。
- 第三章 提出了基于 Scenic 代码变异的自动驾驶感知数据生成与模型测试方法, 详细介绍了构成技术的3个主要方法,深入展示实现方法的基本流程、关 键科学技术原理、设计算法模型理论与实现细节。
- 第四章 针对提出的方法进行了具体的实验,验证本技术在自动驾驶系统质量保障中的有效性。描述实验方案设计和具体实验过程,分别回答了3个重要研究问题,对于具体实验结果进行归纳和解释,并作出进一步的有效性分析和实验总结。
- **第五章** 对于本文的研究工作和实验结果做了概括总结,并对后续研究内容展望 进行了具体描述。

第二章 相关背景技术概述

2.1 自动驾驶系统

自动驾驶系统 (Autonomous Driving System, ADS) 的构建依赖硬件和软件的 高度集成 [43]。硬件设备,例如车载传感器和通信设备对于自动驾驶系统感知环 境(即自动驾驶系统输入)和系统行为决策,即自动驾驶系统输至关重要。自动 驾驶系统的输入包括摄像头(RGB 摄像头、景深摄像头、红外摄像头、语义摄 像头等)、雷达、激光雷达、GNSS 等传感器数据、地理信息系统数据和网络通 信数据;而输出包括系统内部中间输出的感知、规划信号和最终的电子控制信 息(如车速控制、车道保持、碰撞避免、紧急制动、轨迹控制等)。自动驾驶系 统还依赖智能算法软件,该软件负责分析来自硬件的输入,分析交通场景环境 和驾驶行为,基于给定的输入,借助算法模型生成决策,并转换为控制系统可以 识别的电子信号,最终自动化或者半自动化地完成一系列自动驾驶任务。

从系统框架的角度来看,自动驾驶系统可以分为独立自动驾驶系统 [44] 和 基于智能协作互联的自动驾驶系统 [45]。顾名思义,独立自动驾驶系统表明自动 驾驶功能由独立的车辆内部实现,绝大多数的自动驾驶任务由车载软硬件独立 实现。而基于智能协作互联的自动驾驶系统则可能需要依赖其他互联车辆和道 路基础设施,利用通信交互传输重要数据并执行对应自动驾驶任务。目前自动 驾驶软件系统已经逐渐被应用于多种领域,包括军用无人作战、辅助驾驶与自 动巡航、自动物流机器人等,由于其在交通、能源等领域的广泛应用前景,各国 研究单位和企业都投入了巨量的资源,包括政府交通部门和研究单位、传统汽 车厂商 (奥迪、福特、大众等)、新型智能驾驶企业 (特斯拉、蔚来、小鹏汽车等) 以及专攻自动驾驶核心算法系统构建的高科技互联网企业 (如 waymo、Uber、百 度、华为等),这些机构或企业都提出了自己的自动驾驶发展目标,研究对应解 决方案或关键技术。

在自动驾驶功能算法实现方面,可以分为两类,一类是通过模块化各个功能组件并组合(基于模块的自动驾驶系统)[46]实现的,其架构如图2.1,另一 类是通过端到端的集成系统[47]直接实现的。它们之间的区别在于如何做出决 策,或者是否存在明确的决策阶段/子系统。模块化自动驾驶系统由一系列子系 统组成,如目标检测、风险评估、其他系统预测和规划[48],后者将所有功能无 缝集成到一个系统中,不存在明显的功能边界。端到端自动驾驶是近年来开始 出现的一种技术。端到端自动驾驶有三种主要方法:直接监督深度学习[49]、神



图 2.1: 模块化自动驾驶系统架构

经进化 [50] 和深度强化学习 [51]。端到端自动驾驶系统的可解释性常常难以判定,硬编码的系统结构也会造成缺乏有效的软件安全保障措施,容易出现软件缺陷 [52],相对应的,一般测试此类自动驾驶系统中的组件非常困难。

本文相关背景主要研究基于模块化的自动驾驶系统,围绕组件测试进行调研分析。模块化的自动驾驶系统将车辆定位、环境感知、场景评估、规划决策和 行为控制等关键功能独立化,共同组成一个完整的自动驾驶系统,将自动驾驶 这一复杂任务拆分为一系列的子任务 [53]。相对而言,这类系统因为存在多个 模块之间的数据通信和协作交互,更加容易出现质量问题,一旦其中某个模块 发生问题,可能会影响到与之连接甚至是整个自动驾驶工作流下游的所有模块, 从而导致自动驾驶系统出现安全隐患。同时模块化结构也有其优势所在,多模 块的约束将限制自动驾驶系统出现问题的严重程度,可以利用整体协调纠正少 量模块引入的偏差问题,避免整体系统出现崩溃的问题。而在质量保障上可以 针对不同自动驾驶功能模块开发最优模型,并采用不同的独立测试方法 [54],独 立测试可以充分使用待测组件在各自研究领域最优方案,不会对其他模块和整 体系统产生影响。

论文对自动驾驶领域环境感知相关模型进行了研究,发现当前人工智能主要应用于自动驾驶系统系统中的目标识别与跟踪、车辆轨迹预测方面,针对这些任务,国内外进行了长久的研究,产生了很多垂直领域的成果。

2.1.1 道路目标检测

道路目标检测 [55][56] 是目标检测这一领域在自动驾驶领域的重要分支,该 功能使得自动驾驶系统能够高效的发现、定位和跟踪目标交通环境中的不同对 象,如车辆、行人、道路指示牌、道路基础设施等。道路目标首先需要考虑一般 目标检测的基本任务,如实例分割,图像标注和目标跟踪,除此之外,还有一些 更加细致的应用领域,如,交通标注和红绿灯检测,车牌识别等等。早期的传统 目标识别算法基于手工提取特征,其检测过程可以简单概括为几个步骤:

1. 算法选择目标对象中的感兴趣的领域,选择可能包含物体的区域

2. 针对选择的兴趣区域进行特征提取

3. 对提取到的特征进行检测分类

基于传统目标检测算法的模型有 Viola Jones 检测器 [57], HOG 检测器 [58], DPM 检测器 [59] 等,其中 DPM 检测器是传统目标检测算法实现的模型中性能表现 最为优异的。DPM 模型由一个根过滤器 root-filter 和多个分过滤器 part-filter 构 成,借助边框回归,硬负挖掘技术有效地改善了之前传统算法的检测精度问题。 这些模型几乎都使用金字塔多尺度和遍历滑窗的方式,判断当前检测位置是否 存在可识别目标这一思路。

然而,传统目标检测算法计算开销较大,算法模型运行时间长,同时往往对 于目标物体的形变适应能力差,无法有效检测存在大范围旋转内容的目标,可 能存在多个正确识别的结果,还存在稳定性上的缺陷。随着深度神经网络的发 展,尤其是基于卷积神经网络的技术进步,目标检测算法逐渐转为主要由深度 学习算法实现[60][61]。基于卷积神经网络的目标检测算法主要有两种分支发展 路线,包括基于锚 (Anchor-based)的和无锚 (Anchor-free)的算法。锚在计算 机视觉中指代待检测目标的固定参考点或者参考框,在目标识别中,基于锚的 算法首先预设不同尺度不同位置的固定参考框,这些参考框覆盖了多个目标位 置和尺度,每个框复杂检测识别过程中与其重叠度大于阈值的目标,从而将传 统检测问题转化为检测目标框偏离参考框的距离问题,而不再需要进行逐尺度 逐位置的遍历判断。

物体检测任务通常都被建模成对一些候选区域进行分类和回归的问题,基于锚的目标检测算法又进一步分为一阶段检测器和二阶段检测器。前者会使用滑窗方式生成锚框,目前性能较好的典型算法模型代表为 Yolo-v4[11], Reti-naNet[62]等,而后者则先通过区域建议网络 Region Proposal Network (RPN)对

于锚进行分类和回归,再生成对应的候选区域,计算开销更大,速度慢于一阶段 算法,但是拥有更高的精度,典型算法模型为Fast RCNN[63],Faster RCNN[64], FPN[65]等。基于锚的目标检测算法依赖 anchor 本身,模型的检测效果对 anchor 的各项参数都十分敏感,结果受到锚框数量、重叠度(IoU)阈值、尺寸比、图像 划分区域等影响,同时在训练中资源消耗过大。而无锚的算法一般采用确定关 键点的方式完成检测,相比基于锚的方法,能够减少或者完全避免计算 anchor 带来的损耗,典型模型为 CornerNet[66],CenterNet[67],FCOS[68]等。

2.1.2 自动驾驶行为预测

基于感知能力,已经有很多模型用于实现自动驾驶系统的行为预测功能 [69-72]。高级别的自动驾驶中行为或者说轨迹预测是重要的组成部分,预测的 准确度和效率在确保道路安全方面发挥着至关重要的作用,因为能够准确预测 周围车辆运动的自动驾驶汽车能够更好地规划安全的行驶路线,并未一些预测 的道路情况作出正确的应对。车辆预测任务具有挑战性,本质上,该任务的目标 就是让行为预测模型学习交通环境参与者的行为模式,其中包括车辆与行人等 单位的科学理论(车动力学,运动学),道路信息,交通规则,环境障碍物信息 等。这项任务的困难源于动态交通场景的多面性:一个有效的模型还必须考虑 到这些内容在一个具体的交通环境中是动态的组合在一起的。实现轨迹预测的 算法建模方法主要有基于规则的模型[73,74],通过领域数据和专家知识建立一 系列固定条件的决策模式,该模式规则来源于构建的决策树或者物理仿真理论。 基于选择的模型则为各项行为操作建立概率模型,以最大似然估计给出具体操 作。基于 AI 的算法当前最为广泛研究,主要是利用车辆驾驶的行驶数据进行神 经网络训练,通过学习长期的驾驶数据,获得判断执行多种驾驶行为的能力。

本文重点研究了基于神经网络的预测模型,这类模型依赖大规模的高质量 道路数据,同时对于质量有着很高的要求,包括建模交通环境中的道路拓扑结 构,完善的车辆地理位置信息等。Luo[75]等人关注遮挡物和稀疏数据对于自动 驾驶系统可靠性的影响,他们建模了交通场景的3D鸟瞰图,设计的神经网络模 型执行跨空间和时间的三维卷积,分析来自传感器捕获数据进行自动驾驶目标 检测、目标跟踪和行为预测,并进行联合推理。Sauer[76]等人将交通视频输入 映射到适合在复杂环境中的中间表示行驶,具有挑战性的 CARLA 模拟基准测 试上实现了高达 68% 目标导航性能。Liang[1]等人使用了一种新的结构化地图 表示方法以及交通对象-地图的交互,他们的模型拥有超过传统拓扑道路模型的 感知能力,借助融合网络能够准确的预测多模态运动轨迹。HDNet[45]中使用高 清地图中的道路信息提取几何和语义特征,并作为输入特征,成功提高了3D 道 路目标检测的性能。Liang[77]等人提出的端到端行为预测模型中使用栅格化的 地图信息与传感器信息的结合,共同执行联合感知和轨迹预测任务。

然而,基于神经网络的驾驶行为预测依赖大规模的交通场景数据,目前仅 有一些大型汽车公司和研究机构提供了部分数据集 [78-81],人工采集的成本过 高且具有安全隐患,同时受限于场景因素针对一些极端交通场景的用例也不够 充分,无法描述所有可能条件下的场景样本,路测数据质量无法有效地反映在 训练和测试样本中。仿真模拟为重建感兴趣的交通场景提供了一种廉价的机制, 从而实现了一种更具可扩展性的方法。不过,仿真本身也带来了一些问题,需要 满足一些有效条件,包括有效建模仿真环境,有效传感器数据采集,以及为模拟 验证设计的有效的验证指标等。

2.2 自动驾驶测试

当前的自动驾驶核心技术中,人工智能已经深入到几个重要模块中,在环境感知,场景分析和行为预测上发挥重要作用,尤其是感知相关组件其核心算法几乎完全由各类深度神经网络模型实现 [43,46]。作为支撑这些模型技术的自动驾驶软件本身和其他软件系统一样,会出现各种类型软件缺陷,同时受到人工智能技术本身特征和自动驾驶软件系统应用领域的高安全标准影响,自动驾驶的质量保障要求更加严格。

本文关注的自动驾驶系统属于基于组件的模块化自动驾驶系统,关键组件 包括由传感器和感知处理分析模块组成的环境感知组件,由分析感知后数据进 行决策规划的决策规划模块,以及由控制信号处理和车身硬件控制器的控制模 块,系统接受来自感知组件获得的输入数据,并做出一系列的当前环境、道路目 标行为预测、行为规划等一系列内部分析动作,最终由控制组件输出车辆控制、 人机交互相关的电机信号,如速度控制、转向控制、车辆制动、碰撞避免等。与 端到端的自动驾驶系统不同的是,模块化自动驾驶没有直接在一个系统中进行 复杂任务的划分和处理,模块化算法下的自动驾驶系统其功能存在明确的决策 阶段和不同阶段的子系统、模块化自动驾驶系统具有明确的功能边界。高效稳 定的算法是实现自动驾驶的灵魂,随着人工智能、深度学习的发展,模块化的自 动驾驶系统越来越趋于成熟和流行,利用监督学习、深度强化学习、神经进化等 算法,可以实现不同组件系统功能的独立化和最优化。其优势在于不利用硬编 码进行无缝集成,具有更高的安全性和可解释性,同时,针对模块化的组件可 以设计适合的独立测试方法,进一步保障系统的有效性。自动驾驶测试目的在 于发现系统中潜在的缺陷 [82], 自动驾驶的缺陷来源于多种方面, 包括模型算 法本身的结构设计缺陷,训练测试数据的质量缺陷,车辆本身传感器缺陷等等,

2.2 自动驾驶测试

这些缺陷很可能导致灾难性后果,并阻碍自动驾驶技术转化和产业落地[5]。所 有缺陷中,数据质量缺陷是最为重要和常见的,在基于深度神经网络实现的感 知和分析模型中,数据缺陷往往导致模型无法正确认知当前车辆周围交通环境, 并作出具有安全隐患的判断结果。

2.2.1 仿真测试环境建模

自动驾驶环境感知和场景分析模型依赖大规模的有效交通场景数据进行训 练和测试验证 [83],然而相关交通场景数据样本获取困难,真实道路采集数据人 工标注成本高昂,测试里程要求过大,数据采集周期过长,危险场景人工驾驶法 律法规不完善,且标注质量参差不齐,在现实世界中测试人员很难解决这些问 题。同时,交通场景数据还受到数据来源单一、数据多样性不足等因素影响,不 足以满足软件工程自动驾驶研究人员的测试验证要求。为了解决这些问题,国 内外研究者针对交通场景数据的构造生成技术进行了大量的研究,利用数据变 异、仿真引擎生成和游戏模型渲染等方法构建虚拟测试场景数据,从而获得高 质量的测试数据,将不同的生成测试数据用于自动驾驶模型和数据增强。构建 系统测试场景是一项重要的测试任务,对于某些测试任务,测试用例必须足够 丰富,以覆盖测试样本的状态空间,有些需要在极端交通情况下生成测试样本, 测试系统在这些边界用例下决策输出模型的安全性。

在仿真环境的构建上,主流研究方法包括直接使用传统 3D 建模软件构建交 通环境,如 simulink[84]。很多研究人员使用既有物理仿真引擎和游戏引擎构建 虚拟交通环境,如借助 Unreal Engine、Unity、电子游戏 GTA 引擎完成车动力学 模型和道路环境模型搭建,对应推出典型的仿真器包括 AirSim[23]、CarSim[85]、 Apollo[86]、Carla[24]等。部分研究者提出了利用场景合成的构建技术[30],一 种是直接模拟多种车载传感器物理性质,形成模拟车辆的高保真动态信息并在 虚拟场景中呈现,另一种基于增强现实的场景合成,则主要利用采集的真实交 通环境的历史数据信息,这些数据包括路测车载摄像头、激光雷达点云数据,无 人机、卫星扫描的道路网络结构数据,地图服务商提供的高精确度地理信息数 据等,借助数据组合和合成技术,进行数据维度对齐和信息互补,可以获得丰富 的交通场景。仿真环境还可以基于既有数据进行完整建模,将多种环境数据进 行数据融合,增加交通参与者和对应动态交互信息,这些信息包括车辆、行人、 道路设施自身属性以及相应的动作与响应,共同构建全方位的虚拟环境测试场 景。仿真环境可以为自动驾驶系统测试提供高质量的测试样本,一方面,精心设 计的仿真场景在多样性上具有明显优势,可以为不同功能的自动驾驶任务提供 不同条件下的仿真测试样本,利用这些测试样本可以快速有效地进行软硬件的

测试与验证工作。另一方面,可以充分考虑特定需求下的场景仿真要求,生成真 实路测难以复现的特定交通场景,例如指定道路特征和交通参与者动作的极端 交通场景,以检测目标自动驾驶系统的分析和决策边界。当然,相对的,这些虚 拟交通场景仿真方法的测试效果受到仿真引擎本身的性能影响,不同引擎或仿 真技术下的车动力学、静态环境仿真和动态环境仿真能力存在差异,由于渲染 器、传感器模型和材料属性的不同,容易造成仿真测试数据在图像像素上的外 观差异。同时,还存在构建场景与现实不匹配的内容差异,表现为违反真实场景 客观规律、场景多样性不足等问题。因此,仿真测试中对应的仿真测试效果也会 出现偏差。

Zhang 等人 [28] 提出用于自动驾驶车辆系统算法测试验证框架 RoadView, 该框架使用图像序列和地理信息数据进行场景建模,将摄像头数据与 GPS 视 点位置序列数据结合获得视点图像序列,再结合道路 GIS 数据进行交通场景的 三维结构定义,为自主车辆软件系统及其各种算法的测试和验证建立模拟环境。 Zhao[9] 等人基于真实交通场景数据,以图像序列方式模拟自动驾驶车辆的动态 行为,他们在模拟的包含交叉口的典型交通场景下测试了自动驾驶车辆的多种 行为。Li 等人 [30] 提出了增强自动驾驶仿真方法 AADS, 这是一种有效的自动 驾驶场景数据扩增的技术,能够通过模拟交通流增强真实交通场景图像,相关 方法能够使用来自激光雷达和摄像头的路测采集数据,根据其中的车辆轨迹进 一步生成合理的模拟交通流,最后将这些数据进行合成获得各种交通参与者的 组合测试数据,该方法增强的扩增图像具有良好的质量,在真实度、视角多样 性、图像注释丰富度上相对于原始图像来说具有明显的提升。Zofka等人 [29] 设 计的 SleepWalker 框架通过混合虚拟激光雷达扫描结果和真实环境中的传感器数 据,构造了与真实传感器数据构建场景相似的测试场景,利用建立的混合现实 环境模型,在相似的场景中测试自动驾驶功能,该框架成功地添补了道路实测 和基于模型的模拟测试之间空白。他们还将不同的先进模拟方法进行结合[87], 创建更多和更多样化的逼真测试场景,在不同的测试级别上支持自动驾驶的开 发验证工作。Google 的 Waymo[88] 采用模糊化技术变异和组合虚拟交通场景中 的参与要素,以创建用于支持自动驾驶验证的丰富交通场景。Musat等人[31]则 利用提出的模块化数据生成框架,组合不同的天气条件组件来模拟实际交通场 景,他们的系统可以将样本向不利于自动驾驶行驶的场景进行数据增强,有效 地检测了不同自动驾驶感知模型在场景目标识别和语义分割任务上的性能。

研究者们基于构建的的虚拟仿真场景进行大量测试工具和框架开发。Sippl 等人 [89] 提出了分布式的实时交通仿真框架,该分布式仿真框架主要目标在于 对于特定城市交通场景进行整体模拟,集成了行人模拟器和目标车辆模拟器,能 够与不同仿真实例的对象进行交互,可以用于广泛的城市交通场景仿真,来测 试自动驾驶系统。Szalay等人 [25] 基于 Unity 引擎设计了一种新的自动驾驶仿真 测试方法,实时地模拟自动驾驶车辆周围的实际交通情况,使用的实时微观交 通模拟器保障了交通流量的真实性。Gambi 等人 [90] 设计的仿真框架从自然语 言的车祸报告中构建虚拟车祸场景,在模拟的目标车祸现场进行仿真测试。在 他们的另一项研究工作中 [26],还将基于搜索的测试技术与游戏程序内容生成 技术进行结合,从而实现自动创建对于自动驾驶系统具有挑战性的虚拟场景,在 原型软件 AsFault 中, 他们使用该方法生成多种虚拟道路场景以测试自动驾驶车 道保持功能,并成功发现了车道偏离过大的现象。Li[91]等人提出的测试框架整 合了基于场景和功能的自动驾驶测试方法,框架结合了两种测试的优点,利用 重新定义的自动驾驶语义图给出了用于设计测试用例和评估测试结果的定量方 法。Vishnukumar[92] 等人结合机器学习和神经神经网络,将自动驾驶各个开发 阶段的独立系统的测试验证与整体系统测试验证连接起来,具体流程中首先进 行模拟测试,深度神经网络在给定虚拟场景测试初始用例,接着生成覆盖测试 状态空间的多种测试场景组合,并进一步生成多种虚拟测试的测试场景、测试 用例和边界条件,在完成虚拟测试后,真实测试验证将继续虚拟测试中发现的 关键和重要测试内容。

2.2.2 数据驱动测试

数据驱动测试是自动驾驶测试领域的重要分支,是针对特定自动驾驶模型 或系统的有效评估测试方法。由于当前自动驾驶核心组件中深度学习模型的广 泛应用,模型数据依赖的特征十分明显,利用大规模的测试样本可以有效检验 目标自动驾驶模型的可靠性。一些特定的测试数据,则可以发现待测模型在复 杂输入空间中的决策边界,从而发现模型特定场景下的输出变化。数据驱动测 试的核心在于构造有效的高质量测试用例,这些测试用例数据应当针对需要测 试的垂直应用领域采用不同的评估与生成标准。例如,面向自动驾驶多样性场 景测试用例、自动驾驶模型处理的边界极端测试用例,其数据生成和模型测试 方法存在差异。数据驱动测试需要解决几个重要的测试验证问题:

- 如何生成模型敏感的有效测试样本,用于进行测试的生成数据质量如何评估和控制是需要解决的第一个问题。
- 如何基于生成的测试样本设计可解释的测试流程和测试方法,深度学习模型的低可解释性导致测试数据输出难以使用传统软件测试验证标注进行 判定,必须设计合理的能够有效反应模型测试性能的指标和方法,明确显 示有效的测试结果。

2.2 自动驾驶测试

测试结果对于自动驾驶开发人员和软件测试人员的作用和意义,测试的目的不仅在于发现问题,有时还需要针对自动驾驶模型提供有效的帮助,例如如何引导进行数据增强与模型增强。

研究者们在自动驾驶的数据驱动测试中,进行了广泛的研究。当前,针对自动驾驶系统,典型的数据驱动测试是进行交通场景数据生成,包括图像数据、激光雷达数据、以及复杂交通场景数据对应的测试数据生成方法。同时,还针对测试验证方法的有效性问题,提出了多种测试指标。

Pei 等人 [14] 提出了用于深度学习系统的白盒测试工具 DeepXplore, 他们 在研究工作中提出最大化神经元覆盖率和尽可能生成使不同深度神经网络模型 出现不同行为的测试用例,从而实现发掘潜在的极端测试场景,探索输入空间 中不同模型的决策边界,他们使用该测试技术在自动驾驶的图像分析转向判断 模型上进行了测试,并发现了现有模型的错误行为,最后还使用生成的测试用 例进行了模型增强,结果表明对应神经网络的性能得到了有效提升。Tian 等人 [20] 提出并设计的 DeepTest 工具中, 使用了多种数学变换和图像处理技术, 对 于自动驾驶深度学习模型的图像数据添加了一系列的变换方法, 仿照自然界的 雨、雾、照明等驾驶条件,自动化生成多种扩增数据,从而测试深度学习模型 的输入边界,在具体的测试方法中,还利用了神经元激活数量的指标进行测试 用例生成的引导,最终在实际测试中发现了目标深度学习模型中大量潜在的致 命错误。Zhang 等人 [21] 提出了基于深度神经网络的无监督测试框架 DeepRoad, 该框架由一个变异测试模块(用于定义变异生成图像与原始样本的一致性关系) 和输入验证模块(使用图像相似性验证输入图像)组成,能够自动合成大量的不 同交通驾驶场景,该框架避免使用影响图像真实性的缩放、旋转等规则,在补充 DeepXplore 和 DeepTest 相关图像变换方法外还使用了基于对抗生成网络技术生 成风格变异的图像样本,该框架成功在三个有名的模型中发现了数千个不一致 的行为。Zhou[93] 等人提出了一种在现实世界测试自动驾驶系统的新角度,利 用对图像的数字和物理扰动方法生成对抗输入,并诱导转向模型产生不同的决 策输出,同时,文章针对转向模型输出提出了新的指标来评估扰动测试有效性。

使用基于搜索启发式的测试数据生成是一种有效的测试数据获取途径。Li 等人[17]出了一种名为 TACTIC 的测试框架,采用基于搜索的方法来识别由图 像到图像转换模型生成的关键环境条件,利用图像到图像风格转换生成环境条 件丰富的测试样本,该框架使用临街环境条件生成的驾驶场景十分逼真,而且有 更高概率导致自动驾驶深度学习模型输出错误行为。Hussain 等人[27]提出了一 种基于自动编码器和时间序列分析的自动驾驶行为预测异常检测系统,能够预 测由于意外环境导致的自动驾驶模型中潜在的不一致行为,方法通过在 Udacity 模拟器中创建意外的驾驶条件来预测不一致行为,该系统相对 DeepRoad 在确保 自动驾驶安全方面存在性能优势。Calo等人 [22] 使用基于搜索的测试来寻找避 免自动驾驶出现碰撞的系统配置, 在测试过程中, 使用顺序方法和组合方法, 由 路径规划器分别生成用于测试的短期路径与控制命令。结果表明,基于组合的 搜索测试方法更加容易找出目标系统配置。Arcaini 等人 [94] 介绍了驾驶特征模 式的概念,并基于此概念提出了两种基于搜索的方法(针对单个和成对的驾驶 特征)来查找出现此类模式并使其持续时间最大化的场景。实验结果表明,这些 测试方法可以有效地找到路径规划器生成路径的场景,其中不同的驾驶特征按 照指定的模式出现。Hauer 等人 [95] 关注如何设计用于测试自动驾驶系统的适 应度函数,以支持基于搜索的测试技术,他们提供了模板来确保自动驾驶场景 对象在空间中的正确定位,并能够搜索系统离开其安全操作范围的场景。在自 动驾驶的碰撞避免场景中、使用该模板的启发式搜索技术成功找到了所需的目 标测试用例。在 Av-FUZZER[96] 中,研究者通过扰动交通参与者行为来构建潜 在违反交通安全的测试场景,利用遗传算法最小化自动驾驶系统在预测轨迹上 的安全性,作者设计了一个局部的模糊器,在观察到极有可能存在安全危险情 况的区域中增加了局部最优值的利用。该工具成功在 Apollo 平台上发现了五种 不同类型的自动驾驶安全违规行为。

2.2.3 传感器模拟

在图像感知领域,还常常采用对于传感器数据的有效模拟来进行大规模的 核心功能测试。Pereira 等人 [97] 很早就提出了一种基于耦合机器人系统和交通 模拟器系统的自动驾驶模拟框架,在交通模拟系统中创建虚拟城市交通网络,而 在机器人系统中,利用传感器模拟和控制器模拟,动态地获取虚拟环境下的交 通场景图像和其他数据,该仿真测试工具能够支持多种类型的现实交通行为,并 基于传感器和控制器模拟对于自动驾驶系统进行分析。Tuncali 等人 [33] 的 Sim-ATAV 框架支持在虚拟环境中执行自动驾驶测试,利用从物理仿真模型中的 3D 模型构建交通场景,并渲染为 2D 的图像作为感知系统的输入,利用该框架,可 以模拟多种交通场景的摄像头输入,包括一些极端的边界化测试用例,可以通 过对参数化的车辆与行人数据进行调整实现。Dreossi 等人 [34] 设计了一个自动 驾驶图像生成器,可以在低维图像的子空间中采样可修改内容并进行图像合成, 与对抗性生成的策略不同,该模拟的目的在于用于感知系统输入的高逼真度图 像数据,均匀覆盖图像可修改空间并主动添加扰动,而不会仅仅对已有图像进 行扰动修改,作者还设计了一整套的采样方法,用于从目标待测卷积神经网络 提取特征的图像中寻找有效的可修改点。Uricar 等人 [32] 针对自动驾驶中常用 的广角鱼眼相机数据进行了变异和扩增,扩增算法基于对抗生成网络 CycleGAN 和弱标签学习,在目标摄像头数据上,使用特定污染源添加随机污染模式,实现 了对摄像头数据的不透明污染数据扩增,Yang 等人 [40] 使用一种简单地传感器 模拟扩增方法,使用有限的初始车辆的雷达和摄像头数据进行数据变异,该方 法首先基于纹理映射面元将这些数据中重要信息进行提取(交通参与对象 3 位 位置信息、场景特征信息),对于保存后的数据,利用对抗生成网络变更其中对 象位置和方向,并在网络中重建场景。

除了 RGB 摄像头和景深摄像头的图像数据模拟,还有其他类型的传感器 模拟,例如针对激光雷达的点云数据模拟、针对汽车位置记录仪的行驶轨迹模 拟等,这些方法可以用于生成多种类型的自动驾驶模型测试数据,以帮助模型 取得较好的泛化性能。Zhang 等人 [37] 针对轨迹数据模拟提出了一种新的策略, 通过基于策略梯度目标对抗性地生成扩增的轨迹数据,相对基于变换的数据扩 增,模拟生成的增强轨迹数据能够有效地提高强化学习模型的泛化能力,在该 策略中,为了保障模拟数据效果,作者还设计了一个将原始轨迹数据与生成轨 迹数据混合的算法,以减少扩增数据的偏差。Chen 等人 [35] 在 PointMixup 中 优化了传统差值进行的激光雷达点云数据增强方法,通过在两个点云数据之间 路径函数的最佳分配生成新的模拟点云数据,该方法在点云域中引入了基于差 值的正则化器,通过对于点云数据中的噪声扰动和点值几何变换生成了大量有 效测试数据,以验证自动驾驶点云感知模型的鲁棒性。Mamun 等人 [98] 归纳 了一种基于模型驱动工程(MDE)的传感器管理方法,使用逻辑编程语言将车 辆传感器配置转化为给定约束下的领域模型,方法首先模拟传感器信息以创建 多种交通情景, 配置并生成对应场景的传感器设置, 并在 3D 仿真环境中进行 系统验证。Rivero 等人 [36] 模拟了多种不利天气下的激光雷达点云数据, 用于 测试高级驾驶辅助系统和自动驾驶系统,他们的数据主要通过对原始点云数据 中可能出现前窗喷水区域的模拟来构建,利用点云数据合成,使得扩增测试数 据在点云强度、回波数和遮挡方面匹配真实场景,作者还利用增强数据进行了 模型增强,结果表明该点云扩增方法可以广泛应用于不同的点云目标检测和分 类模型中。Yu 等人 [38] 则模拟了自动驾驶系统传感器数据在不同温度和电路 状态下可能存在的不同,通过将标称条件下记录的图像数据与统计电路模型结 合,生成极端交通场景测试数据,该方法配合基于图映射的马尔科夫链蒙特卡洛 算法,采样最少的测试数据,并尽可能准确评估自动驾驶在罕见情景下的故障率。

2.2.4 领域编程语言与自动驾驶测试

直接对测试数据进行变换处理的测试样本生成技术受到处理方法本身的约束,学术界对于处理后的变异数据的有效性也存在一定的争议。相对而言,使用 虚拟仿真构建场景能够构建较为真实的交通测试场景,先进行建模再生成的特 定数据质量具有保障。然而,直接操作模拟器或者仿真引擎 API 进行数据生成 效率较低,在构建交通场景时具有一定的技术阻碍,为此,学术界提出了一些 用于构建自动驾驶场景的领域特定编程语言。这些领域特定语言一般语法简介, 可以很方便的使用简单地代码语句构建丰富的自动驾驶道路场景。借助领域特 定语言支持的模拟器接口,可以获得多样化的场景数据用于测试和验证自动驾 驶模型。与直接使用硬编码方式生成测试图像相比,使用重新设计的自动驾驶

- 提高交通场景的设计开发效率,使用领域特定编程语言可以用于高效的构 建抽象场景模型,在代码编写中缩小代码冗余结构,抽取有效的公共代码 进行内部封装,减少了构建目标交通场景过程中的复杂工作。
- 2. 便于不熟悉特定编程语言的软件工程人员和专家进行测试场景设计和测试数据获取,软件工程人员可以使用较为熟悉的方式进行复杂交通场景编写,而在自动驾驶的垂直应用领域下,领域编程语言可读性一般较高,同时语法直观简洁,相关领域专家可以使用更加简洁高效的领域编程语言进行专业知识转化和测试系统设计。
- 可以在语言中添加一般编程语言不具备的新特性,以适应自动驾驶系统测试,例如添加面向新的场景仿真引擎的易用接口、支持并发、对于场景构 建过程中的代码优化等。

Fremont[10] 等人开发了一种特定驾驶领域的场景描述语言 Scenic,该语言的语法简化了编写复杂交通场景的任务,并支持使用专门的采样技术,以生成有效的交通场景。该语言重视定义场景中的环境配置和空间对象配置,可以定义具体场景的分布,并使用类的概念定义交通参与者及其属性,允许用户以声明的方式为具体的场景构建添加对应约束,并为场景中对象的几何关系提供丰富复杂的非线性表达方式。作者还使用 Scenic 和开发的 Grand Theft Auto V 游戏引擎接口生成交通场景图像,在目标检测模型上使用渲染生成的图像进行训练和验证。同时,作者在后续工作中 [99] 使用 Scenic 定义场景和具体的安全约束,生成形式化模拟的算法测试用例,并评估了构建的模拟运行数据与真实轨道执行数据的偏差,证明了该基于场景的自动驾驶安全性测试方法的有效性。

Majumdar 等人 [41] 提出的 Paracosm 同样是一个专为自动驾驶场景模拟设 计的框架,支持用户以编程方式描述具有特定特征的复杂驾驶场景,在场景中, 可以为不同的参数配置,如使用的道路布局、场景环境条件、交通参与对象的 反馈和响应时间进行个性化的调整。作者使用基于组合测试的方法,对于不同 的参数配置进行了模糊测试,同时还定义了对应的测试覆盖概念。Witsch 等人 [100] 也提出了一种特定语言,用于在自动驾驶场景中建模以进行程序验证行为, 从而协作型自动驾驶车辆的安全问题。作者将该语言从机器人领域扩展到自动 驾驶领域,并展示了在该领域的实际实用性和效率。

O'Kelly[42] 在提出的自动驾驶车辆 CAD 工具链中提供了一种场景描述语 言来创建具体的驾驶场景,该语言可以定义交通场景中的不同代理对象以及实 际的道路拓扑结构,支持在度量时间逻辑中形式化地定义正确性约束。描述完 成的驾驶场景在进行可执行检查后,将会传递给具体的自动化测试数据生成工 具,该工具对于目标场景进行用例生成,可以发现自动驾驶系统违反规范的行 为,并提高测试数据在输入空间中的覆盖率。

领域特定编程语言极大地简化了自动驾驶测试场景构建的流程,使得复杂 交通场景能够被广泛地生成并用于自动驾驶模型测试任务中。不过,高效的测 试方案还依赖如何具体编写和使用这类语言,需要搭配有效的场景构建策略和 数据生成模式,才能保障在基本语言代码本身的可靠性,因此如何保障构建场 景的脚本语言本身的质量也是影响生成测试数据的重要因素。

2.3 本章小结

本章节主要介绍了与研究课题相关的自动驾驶系统、交通环境感知与车辆 行为预测模型算法及相关使用技术。介绍了测试自动驾驶系统的关键测试技术, 归纳了当前国内外自动驾驶质量保障的主要研究方法和研究进展,针对后续技 术与验证模块使用的自动驾驶感知和行为分析模型做了简要的分析和介绍。本 文还重点研究了使用自动驾驶领域特定编程语言进行交通场景设计和测试数据 生成的技术进展,分析其相对于传统图像变换的测试数据生成的优势与不足。通 过调研发现,当前自动驾驶系统测试主要依赖数据驱动测试,其中基于数据变异 和虚拟技术的测试数据生成技术研究是一大热门,但是现有的很多交通数据生 成技术存在缺陷,例如对于交通图像数据的直接变换(像素值变换,矩阵变换) 等存在潜在的图像真实度问题,而一些基于虚拟引擎构建的仿真数据,其生成 逻辑常常以随机化或默认的分布采样生成模式实现,缺少生成模式的引导。因 此,本文工作所提出的借助变异脚本和渲染引擎生成数据的这种方法具有一定 的价值,一方面借助脚本中的重要约束条件与引擎生成的客观规则避免了不合 理图像的生成,另一方面,变异算法为不同场景的生成提供了有效指导,从而可 以构造多种高质量的场景数据。当前针对自动驾驶模型的测试仍然处于发展阶 段,尤其针对一些现有的组件模型测试理论研究相对较少,本文以生成的变异 扩增数据为测试样本,提出的面向自动驾驶感知相关模型测试方法将会为相关 质量保障研究提供有效参考。

第三章 基于 Scenic 的自动驾驶交通场景数据生成与模型测 试技术

本技术将 Scenic 脚本作为构建多种变异场景的待变异对象,设计变异方法, 并基于变异 Scenic 脚本生成的变异交通数据提出面向自动驾驶模型的测试验证 方法。图 3.1显示了本技术的整体架构。首先使用该语言构建了用于后续变异的 典型交通场景,并分析和提取 Scenic 脚本代码文件中可以进行变更的内容,设 计了对应的变异策略。接着,研究了将变异脚本用于生成交通场景图像,编写 了用于引导变异脚本生成的 8 种脚本变异规则,为变异前后图像样本建立映射 关系,以解决无标注信息变异样本生成与测试验证问题,提出了基于规则的交 通场景生成与目标检测模型测试方法。最后,还将变异样本进一步用于测试验 证行为分析模型,提出了基于变异后动态场景、分析工具 VerifAI 和车辆行为轨 迹模型的动态验证方法。具体的技术架构主要包含 3 个模块。首先是基于 Scenic



图 3.1: 总体技术架构图

的场景构建与变异方法模块,参照自动驾驶典型环境特点,创建了以 Scenic 语言编写的测试场景构建脚本,包括 5 个基础种子场景与 4 个复杂种子场景,并基于 Scenic 语言特性,分析了 6 种代码可变异内容和 18 种具体的变异模式,将这

些变异模式应用于新的环境构建脚本的生成,在 Carla 模拟器中进行大规模交通 仿真图像数据生成,这些生成数据模拟了不同的道路场景。接着是基于变异规则的模型测试模块,为变异脚本与原始脚本建立基于变异规则的映射关系,利 用具有规则映射关系的变异前后样本,对自动驾驶目标检测模型进行测试与模 型增强。最后是基于 VerifAI 的模型测试模块,将变异 Scenic 样本的动态测试数 据进一步用于分析和验证行为预测模型,实现测试和验证自动驾驶行为预测模 型轨迹预测任务上的性能。

3.1 Scenic 场景代码构建方法

3.1.1 Scenic 预处理与代码规范化

本技术借助 Scenic 首先构建了一系列的经典场景脚本。作为一种概率编程 语言, Scenic 其语法和语义旨在直观地建模和生成动态和交互式代理行为,可 以被用于生成针对感知系统,尤其是面向自动驾驶、机器人的基于人工智能的 感知系统。具体来说, Scenic 考虑了训练感知系统来处理罕见事件、在不同条件 下测试其性能以及调试故障等问题。Scenic 体现了概率编程语言如何帮助解决 这些问题,方法是开发人员指定编码有趣类型输入的分布,利用 Scenic 进行场 景模拟,并对它们进行采样以生成专门的训练和测试集。这种语言可以用于网 络-物理运行机制下的感知系统和机器人系统,借助丰富的仿真模拟器接口来编 写环境模型,这是任何形式化分析的基本前提。Scenic 关注的系统,如自动驾驶 汽车和机器人系统,其所处的环境是一个场景——即物理对象和抽象代理的配 置。作为一种概率编程语言, Scenic 允许分配场景特征,并对描述的不同分布的 交通场景施加软硬约束声明。UC Berkeley 的团队还开发了从生成的分布中采样 的专门技术,利用了 Scenic 的特定领域语法提供的结构。

为了将 Scenic 代码应用于后续的研究,在不同模型测试方法上提供了不同的数据类型,对于所有 Scenic 脚本文件和代码做了规范与预处理,本文在 Scenic 版本 2.0.0 基础上,调整了部分 Scenic 具体实现,同时舍弃了部分特性。论文中将 Scenic 代码规范为以下格式,见表??,其中对于基本场景,Scenic 原始脚本允许参数定义变量与额外参数,交互行为定义,场景要求项为空,在本文中,将会对这些部分相关代码进行重构:

(1) 构建静态图像的脚本文件中将 ego 对象设定为 Car,同时以包含初始速度的行为 behavior 进行替换,以避免单步渲染因无终止信号而永远运行。

(2) 对于目标 ego,不使用默认的 Carla 空间关系,在 Scenic 脚本中定义好 所需的空间关系内容。



图 3.2: 同一份 Scenic 代码生成的多种场景

(3)一些特别场景特征,如天气 weather,使用均匀采样方法进行随机选择, 并在 Scenic 脚本中显式编码。

(4)包含交互和约束的复杂场景,代码将不使用渲染时间,而是监听代码中的 terminate 语句规定的终止条件触发终止渲染操作。

采取这样的预处理和规范目的在于减少因 Scenic 随机化带来的生成数据的 差异, Scenic 在一些参数缺省的情况下使用默认的环境构建方法, 即使使用同一 份代码也会会产生天气、交通环境、交通参与者的极大差别, 例如图 是根据一 份 scenic 语言生成的抽象场景, 然而其所处的交通环境具有很大的差别, 在可 视化的传感器数据中, 如 RGB 摄像头、景深摄像头、语义分析摄采集到的数据, 会因为环境变化过大产生同样差异巨大的特征, 这对于后续研究是不利的。

表 3.1展示了本文中使用的 Scenic 脚本的模板:

对于一个 Scenic 脚本,该程序一般代表了一个抽象的感知场景,它定义了 初始状态的分布以及场景中对象的的行为。具体场景是从这些分布中采样的场 景,并为初始状态和行为分配了具体值。不过值得注意的是,Scenic 程序每一次 通过模拟器渲染并采样的过程是随机化的,因此每次通过 Scenic 脚本获得的渲 染场景会存在差异。Scenic 要生成场景,需要与模拟器接口进行配合,这些接口 依赖目前场景仿真领域通用的仿真组件,这些仿真组件能够将连接的 Scenic 程 序中对应的场景构建脚本渲染为具有高质量的逼真场景,输出静态场景数据,如 图像、多帧动画,以及其他模拟器支持的场景和交通信息数据,如实时的二维位

表 3.1: 使用的 Scenic 脚本模板

代码模板

#本技术所有 Scenic 脚本使用如下模板进行统一处理 #基本地图与模型定义 param definition model definition # 变量与额外参数 variables definition param definition # 交互行为定义 behavior agent(s): expr #场景与空间关系定义 attribute(s) definition agent(s) with expr #场景要求 require expr

置参数、车辆方向和速度参数等。Scenic 本身支持多种模拟器,包括 DeepGTA V, Carla, WeBot 等,不同的仿真模拟器在使用上存在一定的差别,在本文中使用的是 Carla。

在后续研究中,首先确定了 Scenic 脚本建模中关注的**基本数据、描述符与 算子、声明、模型定义**四类内容,规范了 Scenic 代码重构方法,修改 Scenic 源 码并提供了更多的天气属性和交通参与者模型的支持;接着,参考了国内外在 交通场景中的规章和典型交通环境信息,构建包括汽车、卡车、行人、自行车等 主要道路目标的 5 种基本种子场景脚本和包含交通参与者动态行为的 4 种复杂 场景,并描述了基本的构建方法。这些任务的有效进行离不开本部分的代码预 处理与规范化工作,可以说,本部分的内容为后续场景代码构建的效率和质量 优化提供了有效的帮助。

3.1.2 Scenic 脚本建模

本论文首先在概率编程脚本语言 Scenic 基础上,提出了一种针对 Scenic 脚本的解析变异方法。首先,在选择合适的模拟器基础上,分析该模拟器对应 Scenic 语言固定格式语言文法和语法规则,该语法规则将被用于进一步的解析工作;接着,在语法确定的情况下,将其扩展为通用的语法模板,重新实现自定义语法解析;在实现了语法解析功能之后,重点分析代码可变部分。就可以为下一步骤的脚本变异算法和实验提供支撑。语法解析的目的在于抽取 Scenic 脚本中重要的信息内容,以判断当前脚本可以执行何种变异策略,通过此代码,可以借助 Scenic 支持的模拟器进行有效的物理引擎渲染,从而据此生成一帧或者多帧的场景图像,或是一段动态的仿真场景数据。代码中包含很多用于构建交通场景重要特征的参数,对象,方法或者变量,例如关键的场景地图、交通参与者、行为函数、天气参数,而通过脚本解析,可以将其中一种或多种关键对象值或者关键属性解析出来,并支持后续的变异与分析工作。

原始 Scenic 语法简单高效,例如图 3.3就是使用一个最为基本的 Scenic 语 句 *Ego* = *Car* 生成的,在代码文件中除了基础的模块和参数定义外没有添加其 他场景构建语句,本例中生成的车辆,其使用的车辆模型、所处的位置以及环 境特征等重要参数都是没有确定的,但是 Scenic 支持以默认缺省的方式构建场 景,因此并不会出现问题。使用这种简单的场景构建方法主要有两个弊端,一是 难以定制化具体需求的环境特征, Scenic 将完全以随机化的方式进行场景渲染, 即使是同一个脚本,其仿真渲染后的场景差异存在过大的情形;第二个问题在 于部分省略的写法,不利于变异算法的设计,例如缺乏明确指定场景天气特征 或模型参数,导致变异算法需要增加无意义的匹配算法。

为了减少该类问题,构造适用的场景脚本,本文在 Scenic 基本的脚本构建方 法上省略了一些高级用法,例如自带的针对给定场景中手动变异的语句 mutate, 因为该语句提供的变异能力不足,同时受到脚本硬编码困难、依赖指定标识符 等具体实现方面不足的限制,在官方提供的样例库和相关研究中也很少涉及,因 此避免使用这些内容。具体来讲,本部分主要进行了以下工作:

- 确定了编写和变异后的 Scenic 脚本中使用的基本语言规范,如第三章第 3.1 节所述,重点关注并使用了特定的描述符与算子语法构建脚本和生成 变异场景样本,并编写了用于后续使用的基本代码模板。对于一个 Scenic 脚本代码数据,关注以下一些重要内容:
 - 基本数据 Scenic 中有很多数据类型,包括标量 Scalars (速度,角度值,距离),向量 Vectors (相对位置,空间偏移),方向 Headings (空间方向),


图 3.3: 使用原始 Scenic 渲染得到的一个场景, 该 Scenic 代码仅使用了 1 行代码 进行场景定义

以及向量场 Vector Fields 和 Regions 等等。本文在研究中发现,针对向量场和区域的变异及其不稳定,这两种类型在 Scenic 中应用较少,存在一些随着 API 变更产生的 bug,尝尝会导致 Scenic 脚本出现不可预知的错误。因此,提出的变异方法主要针对其他 3 种类型的数据进行,其他两种,在一些特殊情况下使用。

- **描述符与算子** 算子 Operator 可以操作 Scenic 提供的对象。基于对象的数据类型提供不同的算子,本文主要使用的算子语法见表格 3.2:
- **声明** 声明是 Scenic 脚本中比较独立的部分,包括导入模块,基本参数,环 境约束要求和终止语句声明。基本参数涵盖使用的地图、场景天气 特征、额外的变量参数设计,环境约束则是在仿真模拟中,要求在迭 代 Scenic 脚本过程中,模拟器将符合给定约束条件的场景视为有效 场景。本研究中,将会主要针对基本参数进行变异,同时,对于环境 约束要求和终止声明,也将会在具体的变异模式中使用。
- 模型定义 环境中参与交互的对象依赖 Carla 提供的有效模型,同样的,研 究中首先对 Scenic 支持的模型库,尤其是车辆模型库进行了补充,将 其未更新的 6 种交通工具模型加入了可定义列表。在不同的变异方案 中,对于模型本身的变异是一个重点,包括增改对象,更改对象模型 类别等。
- 2. 使用显示编码方式重构了 Scenic 代码, 使得一些重要的可变异内容可以被

类型	使用语句	意义
位置/区域说明符	at vector	放置对象于给定坐标
	offset by scalar	放置对象于局部坐标系的给定坐标
	(left right) of vector [by scalar]	按给定的标量距离将对象进一步向左/向右定位
	visible [from (point OrentedPoint)]	在自我的可见区域,或给定点/定向点 (如果给定)的
		可见区域,均匀随机地定位对象
	(in on) region	在给定区域中均匀随机地定位对象
	following vectorField [from vector] for scalar	将对象定位在一个点上,该点是通过从 ego 开始,按
		照给定距离的给定向量场获得的
方向说明符	facing heading	在全局坐标中沿给定的方向确定对象的方向
标量算子	distance [from vector] to vector	从自我到给定位置的距离(或带有可选自向量的位
		置)
	angle [from vector] to vector	从 ego 到给定位置的方向(或带有可选 from 向量的
大户体了	Daint and and (matter Object)	位直) 川 - 人士式老台台卡张不毛到日 - 人士式老叶鱼
巾 /小昇丁	Point can see (vectpr Object)	从一个点或者定问点能沿有到另一个点或者对家
万问算于	scalar deg	角度偏移度数
向量算子	vector (relative to offset by) vector	第一个向量,解释为相对于第二个向量的偏移量
定向点算子	vector relative to OrientedPoint	在定向点的局部坐标系中解释的给定向量

表 3.2: 使用的描述符和算子语法及其含义

检测到。例如, Scenic 默认情况下不会显示编写天气状况,同时每次仿真 渲染过程中调用 Carla API 从给定的几种天气预设值中进行均匀选择,本 研究中,首先扩增了 Scenic 支持的天气种类,加入了7种天气状况下的夜 晚天气,然后使用参数初始化 *paramweather* = *Uniform(weather)*的方式, 从扩增天气集合中均匀采样选择天气,并显式配置当前 Scenic 天气状况。

下页描述了一个本技术使用的一个 Scenic 脚本,该脚本描述并且建模了一种多个交通参与者交互的场景,该脚本片段建模了一个障碍物出现在目标车辆前方的典型交通场景,其中第7至11行定义了一些基本的约束参数,这些变量值来自 Scenic 库中要求定义的必须参数,例如 EGO_BRAKING_THRES HOLD 变量用于后续 Scenic 中检测目标车辆 ego 与环境中设置的其他道路对象是否到达制动阈值距离,并输出判断信号的监听函数 withinDistanceToAnyObjs()的形参取值。第30行至38行定义了当前 Scenic 脚本的空间关系,包括目标车辆的道路 lane,目标车辆初始位置(出生点) spawnPoint,障碍物 obstacle,以及默认目标车辆 ego 本身的初始化。目标车辆 ego 的行为在第15-20行进行定义,首先在try 代码块中将会进行正常的沿着当前道路行驶,当行驶前方出现障碍物且距离达到阈值,为了避免碰撞将会进行制动行为,该逻辑由第19行的中断 interrupt 完成,可以在一个 Scenic 脚本中定义多个中断,或者说交互行为,在本中断行为完成后,目标车辆将继续尝试保持初始 try 中的默认驾驶行为,或者在此过程中继续触发中断操作。第45 行则定义了一些重要的环境约束,类似其他概率编

3.1 Scenic 场景代码构建方法

程语言中的观察语句,以保障生成的场景满足给定的硬性条件,否则视为不符 合要求的模拟场景。

```
# 设置使用的环境参数
  param map = localPath('../../tests/formats/opendrive/maps/CARLA/Town01.xodr')
2
  param carla_map = 'Town01'
3
   param weather = Uniform(
4
      'ClearNoon'
5
6
  model scenic.simulators.carla.model
   ## 基本约束
8
  EGO_MODEL = "vehicle.lincoln.mkz2017"
9
   EGO SPEED = 10
10
  EGO\_BRAKING\_THRESHOLD = 12
11
  BRAKE ACTION = 1.0
12
   ## 关于ego和其他对象的behavior定义区域
13
   # 动作说明: ego车辆在正常沿道路行驶下, 遭遇前车
14
   behavior EgoBehavior(speed=10):
      try:
16
          do FollowLaneBehavior(speed)
17
      interrupt when withinDistanceToAnyObjs(self, EGO_BRAKING_THRESHOLD):
18
          take SetBrakeAction(BRAKE_ACTION)
19
   # 可以为不同对象定义多个behavior
20
   # behavior BadlyParkedPulledCar(laneToFollow, speed)
21
   #
        while(distance from self to ego) > 10:
22
   #
            wait
23
   #
        do FollowLaneBehavior(laneToFollow=ego.lane, speed)
24
   ## 关于对象空间关系定义
25
   lane = Uniform(*network.lanes)
26
   spawnPoint = OrientedPoint on lane.centerline
27
   obstacle = Trash at spawnPt offset by Range(1, -1) @ 0
28
   ## 场景构建
29
   ego = Car following roadDirection from spawnPt for Range(-50, -30),
30
      with blueprint EGO_MODEL,
31
      with behavior EgoBehavior(EGO_SPEED)
32
   require (distance to intersection) > 75
33
   terminate when ego.speed < 0.1 and (distance to obstacle) < 15
34
```

3.1.3 典型场景构建

使用 Scenic 语言进行了典型场景代码编写。参考国外自动驾驶相关领域较 为权威的美国国家公路交通安全管理局(简称 NHTSA)发布的自动驾驶安全 隐患资料和国内交通部发布的《中华人民共和国道路交通安全法》,本技术在 Scenic 提供的 API 和基础样例之外, 重新设计了 Scenic 脚本库。这个脚本库包 含5个基本 Scenic 场景脚本,以及4个较为复杂的交通场景,这些脚本文件称为 种子场景脚本,其语法为 Carla 模拟器的适配做了一定修改。Scenic 脚本库中的 文件编写主要基于 Scenic 支持的多种类型对象和环境构建能力, 借助 Carla 模拟 器接口,重点考察允许自定义类型、位置信息、动作和交互行为的车辆、行人、 环境参数和交通标志对于生成场景的影响。简单种子场景脚本包括5种类型的 静态道路场景图像,这些道路场景构建较为简单,但是包含了汽车、卡车、行人、 自行车和交通指示灯等在内的基本道路交通对象,脚本中不包含复杂的交通对 象交互,适合4.1节中大部分的变异方法,可以很好地用于后续的变异场景静态 图像生成。4个复杂 Scenic 种子场景则分别描述了汽车与汽车、汽车与自行车、 汽车与行人以及多车辆活动的基本场景,这些场景将被用于变异生成动态交通 场景数据,用于测试和验证依赖序列化数据的自动驾驶模型。首先本文使用了 在第3章第1节中提及的 Scenic 脚本规范和 Scenic 脚本代码重构方法,对于构 建的初始场景进行了对应的修改,见算法1其详细的实现原理不再赘述。简单来 讲,通过这些操作,将构建的5种基础种子场景与复杂场景进行了语法格式的 一致化。

最终,构建的种子场景脚本及其对应具体的描述场景如表 3.3,使用算法1代码模板进行了重构,基本种子场景中一般不包含非目标 ego 以外的动作,默认条件下车道,汽车出生点和汽车类型以均匀采样方法从 Scenic 支持的 Carla model 库中选择。

3.1.4 脚本解析与变异对象分析

Scenic 的运行离不开具体的场景脚本,即以".scenic"为后缀名的自定义代码文件。针对已经构建的脚本提出了一种解析方法,结合后续的代码变异算法,使得一份基础种子场景 Scenic 代码可以通过进行变异来构造新的脚本文件。本文设计了面向基础 Scenic 代码文件的代码解析和变异对象分析方法,方法首先将扫描基本代码文件,保存原始代码信息,然后对于原始代码文件中的不同区域块,使用特定的匹配算法提取可变异对象。这种扫描分析方法是自上而下实现的,这与 Scenic 代码本身构造流程是类似的。在构建种子场景 Scenic 样本时,遵循了前文中定义的编码规范,这为提出的线性扫描并执行变异的算法提供了

_

Algorithm 1: Scenic 脚本语法一致化				
1 #将普通 Scenic 脚本文件转换为适用于后续所有研究和实验格式				
2 Input: 脚本路径 path				
3 curScenicCodes ← readFromScenicFile(path);				
4 添加初始化天气,默认为均匀采样获得的随机天气效果				
5 if !hasParam(weather) then				
6 addToCode(curScenicCodes, statements_of_weather)				
7 为不包含动作的目标 ego 添加行驶动作, 防止批处理时无法停止一个				
step 的模拟				
8 if !hasParam(behavior) then				
foreach angent in curScenicCodes do				
10 if typeOf(agent).equals('Ego') and				
11 !hasSpeed(angent) then				
12 addToCode(curScenicCodes, statements_of_speed)				
13 addToCode(curScenicCodes, statements_of_behaviors)				
14 behavior \leftarrow default agent \leftarrow behavior				
threshold \leftarrow val				
终止条件添加				
17 \lfloor terminate_statement \leftarrow 'when distanceToEgoSpawn > threshold'				

		基本种子场景
种子编号	名称	场景描述
1	base_seed1.scenic	基本道路场景,存在一辆默认车辆
2	base_seed2.scenic	包含道路行人的场景
3	base_seed3.scenic	包含同方向车辆场景
4	base_seed4.scenic	包含反方向车辆场景
5	base_seed5.scenic	包含交通路口指示场景
6	seed6.scenic	描述 3 路路口下具有动态行为的汽车的场景
7	seed7.scenic	描述 3 路路口下具有动态行为的卡车的场景
8	seed8.scenic	描述 3 路路口下车辆与具有动态行为的自行车的场景
9	seed9.scenic	描述存在大量交通对象的动态场景

表 3.3: 包含基本语法的种子场景

帮助,该分析方法最终采用以下流程进行可变异对象的匹配与解析:

1. 扫描 Scenic 代码文件基本地图和模型定义区域,保存当前脚本代码文件 中场景地图类型参数 carla_map 使用的地图变量名称、天气和时间参数

3.1 Scenic 场景代码构建方法

weather 下目前使用的均匀采样的天气名称。

- 扫描变量与额外约束代码区,以键值对形式保存当前定义的变量名称与对应值,在后续分析中,一旦扫描到了具体变量名,搜索保存的键值对,将 具体变量使用其真实值进行替换。
- 3. 扫描交互行为定义区域,保存定义的动作 behavior 名称和对应的动作代码。
- 扫描场景与空间关系定义区域,首先搜索是否存在显式的空间关系定义, 包括对象所处的道路、轨迹和目标出生点位置,接着将会扫描所有交通参 与者对象,并将对应的信息以[对象名称,类型,位置,使用模型,动作] 的列表格式进行保存,其中的变量名称在搜索后使用真实值替换。
- 5. 扫描场景要求和模拟终止条件, 替换使用的变量名称为真实值。

利用该分析方法,可以提取目标 Scenic 脚本中的可变异对象。对于基础交通场景脚本,即上文中定义的基本场景种子脚本,本实验关注的是使用这类脚本进行变异后,渲染等到 Carla 中的静态场景图像信息,本方法使用在固定时间点进行渲染场景保存的方式获得对应图片。在图片相关变异对象分析时,需要保障 Scenic 脚本代码中的动态的行为的改变不会影响后续静态图像的获取。举例来讲,在基础场景中实现一个添加卡车的变异场景,需要在对应的 Scenic 脚本中添加对应的卡车对象的代码,本方法期望在新的获得的所有仿真渲染场景数据中存在一张与原始输出图像仅在"车辆周围有没有一辆卡车"存在差异的样本,因此在空间关系定义中,需要保障在保存图片的时间点,两次模拟时保持场景的一致性。这种要求下,本方法不会添加任何涉及车辆交互代码,仅仅保障添加的卡车对象位置尽量处于目标 Ego 车辆周围:仅在交通参与者对象定义代码下,新增卡车对象,使用偏移距离、相对角度和方向定义具体的位置信息。设计了比较简洁高效的模式匹配方法,由于这类脚本只是用于输出较为简单的场景,这种匹配方法相对简洁,在仅希望利用变异 Scenic 脚本进行图像变异输出时十分有效,且不会对原始场景添加过多的动态交互内容。

针对复杂 sc 脚本,即用于后续行为预测模型的测试样本,这类脚本一般描述了较为复杂的道路交互操作或其他约束,因此在模式匹配基础上,还需要通过仔细的检查进行有效性分析,对包含复杂交通条件的 Scenic 样本,进行了代码抽象语法树分析,并利用绘图工具可视化不同 Scenic 脚本代码抽象语法树,以研究其代码构造模式,发现这类代码在空间定义和场景构建上通常基于一定的构造格式,如图 3.4,具有相似的代码结构。例如,在车辆前方添加同样具有行驶动作的车辆时,将首先搜索分析代码中保存的对应 Ego 车辆位置和行为信息中

是否使用了空间关系定义中的内容,如设定的道路段、车道和行驶轨迹信息,一般这类信息以相似代码进行编码,基于搜索到的 ego manuever (定义交通场景的基础类),为卡车对象定义对应的 manuever,才能保障在动态仿真过程中保持与目标 Ego 的联系。基于以上分析流程和分析方法,在变异中采用表 3.4进行具体的变异,部分变异后的效果如图3.5。

在测试一个道路目标检测相关模型过程中,使用包含不同场景条件的测试用例 是有效的,可以测试和验证模型性能。将多种类型的变异方法以随机化方式进



代码1:为添加的转向车辆对象生成meneuvers,构建空间关系部分抽象语法树



代码2:为添加的直行车辆对象生成meneuvers,构建空间关系部分抽象语法树

图 3.4: Scenic 脚本语法解析树分析和可视化

3.1 Scenic 场景代码构建方法

可变异内容	变异方法
地图	默认 param map 替换为随机 5 张地图之一 指定变异 map 名称
天气	默认变异为从 [*Sunset, *Noon] 的所有天气场景中随机均匀采样 指定变异从 Rain 中的所有天气场景中随机均匀采样 指定变异从 Clear 中的所有天气场景中随机均匀采样 指定变异从 Wet 中的所有天气场景中随机均匀采样 指定变异从 Cloudy 中的所有天气场景中随机均匀采样 指定天气名称
时间	指定变异从 Night 中的所有天气场景中随机均匀采样 指定变异从 Noon 中的所有天气场景中随机均匀采样 指定变异从 Sunset 中的所有天气场景中随机均匀采样
场景	修改对象获得的 manuever 中驾驶类型,随机替换为 LEFT_TURN, RIGHT_TURN, STRAIGHT 修改交通路口类型,3way 4way 互换
对象类型	修改同类型 blueprint 为随机模型 移除目标对象 修改为可运动其他类型对象,从 Car, Truck, Bycicle 采样
数值与关系	标量值, deg 添加偏移取模 180, 距离添加偏移量 向量值, 位置添加偏移

表 3.4: 变异内容与变异方法



图 3.5: 执行变异后不同变异脚本仿真渲染结果中采样的近似场景图片

行图片生成,设置用于训练和学习图像特征的训练集 Dataset_train,接着,额外 生成用于进行模型验证的测试集 Dataset_test,统计几种模型测试结果信息中不

同类型的检测问题。

典型的检测问题包括:

- **漏报** 目标检测模型在当前测试图像中出现可漏判,未能识别目标对象类型或置 信度过低。
- 误报 目标检测模型识别出了物体,但是输出了错误的目标类型。

预测置信度偏差 模型预测的总体平均置信度下降。

3.2 基于变异规则的道路目标识别模型测试方法



图 3.6: 基于变异规则的道路目标识别模型测试方法示意图

在实现了基于变异策略的 Scenic 脚本变异方法设计后,可以利用不同的变 异策略,获得多种变异后场景脚本。然而,随机性的变异策略缺乏可靠性,上述 提出的变异方法本身缺乏一定的模型测试评价标准,如何基于变异方法生成具 体测试交通场景样本,如何科学化评估度量样本并用于测试增强真正的自动驾 驶模型是要解决的第二个问题。本文提出针对变异后代码进行有效分析的评估 方法(图3.6),以分析和使用高质量的变异测试样本。方法基于蜕变测试(Metamorphic Testing, MT)思想,仿照蜕变测试中定义等价关系、设计变异算子、验 证一致性输出的思路,针对重要的代码变异定义了一系列的变异规则,这些规则规定了种子场景样本和变异后场景样本之间的映射关系。在本方法中,变异规则将指导一部分的变异样本生成模式,并实现对目标检测模型的有效验证与 增强。

3.2.1 变异规则引导的样本生成

一个基本的变异规则包含规则内容描述,规则分析,预期输入和变异规则 映射关系。首先,针对基础种子场景的变异,定义了8种基本的变异规则,见表 格3.5。

由于生成的场景图像数据针对目标检测模型,本方法将变异规则中的指标 计算定义为与测试种子场景样本与输出中相关指标的数学化约束。例如变异规 则1 描述了在给定目标场景下在目标车辆前方添加车辆的变异模式,*ComputeMut* 使用与输出目标检测类别与置信度的约束条件,此时要求变异前后样本在目标检 测模型下的检测对应类别的目标数量提高,同时,不应当对于目标检测的置信 度产生过多影响,偏差为 $\omega_o bj$,否则说明生成样本在该变异规则下生成图像质 量存在问题,考虑到变异 Scenic 脚本生成过程中对于除目标车辆意外对象构建 的随机性,默认变异前后 Scenic 脚本生成 N 张渲染场景图像,并为约束中的数 字添加模糊化具体约束条件的参数 λ

$$ComputeMut : \frac{1}{N} \sum_{i=1}^{N} (DetecNum(X_{mut}, car)) -DetecNum(X_{ori}, car)) <= \lambda * t$$

$$\frac{1}{N}abs(Predictions(X_{mut}, car) - Predictions(X_{ori}, car) < obj$$
(3.1)

检查基于规则的输出是否满足定义的约束条件,可以帮助判断当前变异规则生成的样本是否合理。目前,设计并引入了4种面向道路目标检测变异规则的计算约束方法,见表格 3.6,这些规则被用于组合生成不同变异模式下的测试样本。

3.2.2 模型测试与增强方法

选择合适的模型进行基于规则的变异样本测试,首先以4.1节中变异样本生成的随机生成的图片对于目标识别模型进行重新训练,以使得模型能够有效学习 Carla 生成的图像场景数据特征。然而,训练数据采用随机化选择训练样本,缺乏有效划分,且数据规模较小,该训练得到的模型的泛化能力和性能包含潜在的风险。本文设计了针对这类包含潜在质量问题的模型的变异测试方法,使

表 3.5: Scenic 样本变异规则设计

规则 1-3

变异内容: 脚本中目标车辆前方添加对象,将导致渲染得到的多组目标检测中,平均目标检测数产生偏差,平均检测置信 度产生对应的偏差 ω_obj

规则分析:变异对象,包括添加和删除对象,目标检测模型识别物体数变化,数量偏差与t相关,同时不同类型像素大小 存在差异,变异后置信度产生变化

Rule1:添加汽车/卡车对象 t 个,与默认车辆距离低于可见距离,平均目标检测数偏移为 t* λ , λ =0.5, 置信度偏差 ω _obj 为 0.1

Rule2: 添加行人对象 t 个,与默认车辆距离低于可见距离,平均目标检测数偏移为 t*λ, λ =0.5, 置信度偏差 ω_obj 为 0.2 **Rule3**: 变更对象模型,平均目标检测数偏移为 t*λ, λ =0.1, 置信度偏差 ω_obj 为 0.1

预期输入:目标检测模型置信度 Predictions,检测类型目标数 DetecNum,原始场景脚本渲染图像 X_ori,变异脚本渲染图 像 X_mut,变异对象数量 t

变异规则映射关系: avg(Predictions(X_ori)-Predictions(X_mut))< ω_obj and avg(DetecNum(X_mut)-DetecNum(X_ori)) <= λt

规则 4-6

变异内容: 变更种子场景中默认天气为 mut_weather, 将导致渲染得到的多组目标检测中, 平均检测置信度产生对应的偏差 ω wea

规则分析:变异天气将会改变获得图像中场景光照、降水积水量,太阳角度,时间,对于图像像素值具有不同的影响,进 而影响目标检测模型置信度,mut_weather 类型参数影响偏差 ω_wea 值,默认 ClearNoon 情况下:

Rule4: 雨天变异下,从默认天气类型 ClearNoon → SoftRainNoon, WetRainNoon, HardRainNoon, 对应 ω_wea 为 0.1, 0.2, 0.3

Rule5: 道路积水变异下,从默认天气类型 ClearNoon → WetNoon, 对应 ω _wea 为 0.1

Rule6: 阳光角度变异下,从默认天气类型 ClearNoon → ClearNight,对应 ω _wea 为 0.5

预期输入:目标检测模型置信度 Predictions,检测类型目标数 DetecNum,原始场景脚本渲染图像 XoriX_mut

变异规则映射关系: avg(Predictions(X_ori)-Predictions(X_mut)) < ω_wea

规则 7-8

变异内容:脚本中标量与位置偏移变异(仅对于相对 Ego 对象的数值),导致渲染得到的多组目标检测中,平均检测置信 度产生偏差 @rel

规则分析: 变异代码中的标量与向量值,添加偏移操作会影响目标结果中距离初始点的对象像素大小,变异后置信度产生 变化

Rule7: 将初始与 Ego 的距离标量 dis 添加偏移 b (-dis<b<dis), 平均置信度偏差 ω _rel 为 0.1(偏移与相对 Ego 初始距离的比 值 +1)

Rule8: 将初始相对 Ego 的某个向量 vec 添加偏移 b, 平均置信度偏差 ω_{rel} 为 0.1(新增与初始向量与 Ego 之间距离的比值) 预期输入: 目标检测模型置信度 Predictions, 检测类型目标数 DetecNum, 原始场景脚本渲染图像 X_ori, 变异脚本渲染图 像 X_mut, 偏移值 b (或 \vec{b})

变异规则映射关系: avg(Predictions(X_ori)-Predictions(X_mut)) < ω_rel

用利用不同规则引导的新的测试集对进行模型测试,基于定义的映射关系与同 一类型变异前后样本测试输出的一致性与不一致性,判断模型在特定种类样本 上的目标检测性能与泛化能力。

基于设定的 8 种变异规则, 分别使用变异 Scenic 脚本在 Carla 下生成新的图 像场景图像, 分析模型性能。在待测模型预测结果中利用约束完成对变异测试 样本的评估之后, 将生成的符合约束的变异样本划分为 2 类, 良好的没有违背 规则的变异样本, 称为为一致样本, 以及在模型预测中出现明显违背规则情况

3.2 基于变异规则的道路目标识别模型测试方法

12 5.0.	—————————————————————————————————————					
约束名	描述	作用				
DetecNum(ret,typestr)	给定预测结果中目标 类别预测数	对象变异: Rule1-3				
DetecAll(ret)	所有预测结果类别数	天气变异: Rule4-6				
Predections(ret, typestr)	单次预测结果目标类	对象变异: Rule1-3				
	别置信度加权均值					
Predections(ret)	单词预测结果所有检	天气变异(除Night类				
	测类置信度之和	型) ¹				

表 3.6: 用于评估变异映射关系的方法

的样本,称为不一致样本。本研究的实验基于一定的假设,一致样本其测试结果 表明对应的变异 Scenic 脚本具有一定的合理性,反应了模型在特定类型数据上 的泛化能力。因此,使用一致性样本的脚本生成新的测试图像,组成的数据集 可能具有在模型训练验证和测试上的价值。将未违反约束的一致数据进行采集, 重新组成新的数据集 Dataset_con。对于出现明显违背约束的变异样本,通过人 工检查变异后 Scenic 脚本的方式,分析是否是因为 Scenic 代码本身在生成过程 中出现的质量问题导致了违反约束的情况,将没有明显缺陷的脚本称为非一致 性脚本,构建的图像组成数据集 Dataset_incon。



图 3.7: 目标检测中 IoU 与检测框示例

使用 Dataset_con, Dataset_incon 分别对于目标检测模型进行再训练,分别 对比模型训练前后的性能指标,尤其是检验非一致性样本对于目标检测性能是 否存在有效的正向增强现象。其中,分析模型指标主要采用了精确率 (Precision)

¹夜晚天气变异样本置信度明显下降,因此在此类型的变异时不会考虑置信度偏差,这一类型的数据也 仅在后续重训练模型的实验中采用作为负样本。

与召回率(Recall)进行计算。精确率和召回率的计算与4个基础概念值计算有关,分别是

- TP(True Positive)即被模型预测为正类的正样本
- TN (True Negative) 被模型预测为负类的负样本
- FP (False Positive) 被模型预测为正类的负样本
- FN (False Negative) 被模型预测为负类的正样本

在一般的模型中,不同样本的归属,可以如表 3.7中所描述的一样进行分类。基

表 3.7: 柞	羊本分类	と 图
	预测	情况
具实情况	正例	反例
正例	ТР	FN
反例	FP	TN

于这些概念,可以计算具体对应的精确率和召回率:

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\text{TP}}{\text{all detections}}$$
(3.2)
TP TP

$$R = \frac{\Pi}{\text{TP} + \text{FN}} = \frac{\Pi}{\text{all ground truths}}$$
(3.3)

在目标检测模型中,基于重叠度3.7 (Intersection over Union, IoU,或者称为 交并比) 定义了多种 AP 与 AR 相关指标。IoU 是目标检测中重要的计算和评价 指标,它表示在目标检测中标记框(即目标的 ground_truth) 与模型检测框的交 集除以并集,不同的 IoU 阈值 (0.50-0.95) 可以得到目标测试结果中的不同结果, IoU 帮助定义常用的 FP、TP 等计数在目标识别中的计算方式,其中 TP 表示的 是 ground_truth 框作出了正确的分类,预测框与其的 IoU 大于阈值,FP 表示对 应不存在的东西做了预测,或者是预测的 IoU 小于阈值,FN 则表示目标检测出 现了遗漏,而关于 TN,在目标检测中通常不进行考虑,因为一张图片中可以画 出无数的框。

$$IoU = \frac{\text{area of overlap}}{\text{area of union}}$$
(3.4)

本文提出的方法基于不同的 IoU,分析了几个具体的目标检测模型性能指标。

3.3 基于 VerifAI 的车辆行为预测模型测试方法

AP 平均准确度,表示所有设置的 IoU 阈值(范围从 0.5-0.95,以 0.05 量级
 递增共 10 个级别)和目标类别下预测平均值。在 Yolo 模型中 AP 实际上
 等同于其他模型中的 mAP 概念,计算方式为:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i$$
(3.5)

即每种 IoU 下预测精准度 P_i 的均值。

- AP50, AP75 IoU 为 0.5 和 0.75 下的平均精确率度。
- AP_large,AP_medium,AP_small,分别根据目标对象大小(像素面积,分别 对应对象大小为[96²,10000²),[32²,96²]和(0,32²])的平均精确率。
- AR 平均召回率,在给定固定数量的检测结果中的最大召回,并在所有 IoU 阈值和类别上求平均值。
- AR1, AR10, AR100, 分别为平均1次、10次和100次检测的平均召回率。
- AR_large,AR_medium,AR_small,分别根据目标对象大小(像素面积,分别 对应对象大小为[96²,10000²),[32²,96²]和(0,32²]),在给定100次检测下的 的平均召回率。

3.3 基于 VerifAI 的车辆行为预测模型测试方法

3.3.1 仿真数据实时采集

Scenic 能够支持使用 VerifAI[101],这是一个用于对人工智能和机器学习模型系统进行验证设计和分析的工具包。用户可以通过定义参数范围,参数化感兴趣的目标系统和数据特征空间,在这个空间中由 VerifAI 使用不同的采样方法,生成多组具体的测试用例。VerifAI 将会使用定义的指标计算方法对每次测试生成具体的测试结果进行分析,判断是否存在违背或者满足预设的系统规范。本论文提出的方法中使用了这一特性,实现了一个可以有效分析和验证生成变异脚本质量功能的模块,使得生成的 Scenic 测试样本数据能够有效测试和验证给定的行为预测模型,并将行为预测模型、VerifAI 工具和变异测试样本用于构建完整的测试流,详细搭建的 VerifAI 模型架构如图3.8所示。

该 VerifAI 模型的 3 个输入为待测试或验证的自动驾驶目标模型,变异扩增构建的 Scenic 脚本场景在 Carla 中采样的传感器和过去历史轨迹数据,以及用于评估分析变异样本 Carla 实际轨迹与预测轨迹差异的的多维度指标定义。在目标模型上,训练并使用了 LaneGCN 模型,作为待验证的自动驾驶行为模型,该模



图 3.8: 基于 VerifAI 的模型测试方法示意图

型能够将 Carla 获得的多帧车辆环境和地理位置信息数据作为一项有效输入,输出多模轨迹预测。

Scenic 脚本场景来源于种子场景变异扩增获得的新的样本库。多维度的定义指标实现上主要围绕 Carla 模拟器可以获得的复杂信息数据。变异后脚本在进行 Carla 模拟时,除了可以保存有效的视觉摄像头的传感器数据,每次进行仿真模拟时,可以将其他有效的仿真信息采集,本研究实现了从模拟器中获取这些关键仿真信息,包括:

- 传感器信息。包括 2D RGB 图像数据,景深摄像头数据和雷达数据。
- 车辆轨迹数据,参与交通交互的车辆历史轨迹数据,即不同时间戳下车辆 的二维位置参数和方向向量。

对于 4.2 中通过评估得到的有效变异样本,本论文进一步采样获得 3 个变异脚本,并继续对这些脚本进行处理,从中筛选可以用于进行目标轨迹检测的动态场景,这些样本是变异后包含动态环境和实体交互的车辆驾驶场景。

3.3.2 车辆轨迹预测模型

本文实现了将 Scenic 变异样本在 Carla 中的轨迹作为输入的车辆轨迹预测, 该功能以行为预测模型 LaneGCN[1] 为基础,将 Scenic 变异脚本在 Carla 运行中 得到的交通环境中的有效信息作为模型重要数据使用。行为预测模型 LaneGCN 是由 Uber Research 提出的用于轨迹预测的深度神经网络模型,是自动驾驶行 业为数不多的开源的行为预测模型之一,曾经在 Argo AI 举办的基于 Argoverse 数据集的运动预测竞赛中取得第一的成绩。该模型的特别之处在于没有使用语 义地图信息,避免了信息损失,而是将交通地图数据组织为 Lanegraph 这种包 含地图结构信息的格式,扩展了图卷积神经网络(Graph Convolutional Network, GCN)以捕捉复杂的拓扑和长距离道路依赖关系,相对复杂的道路拓扑卷积,捕 捉特征的能力有明显优势。LaneGCN 提出并使用了融合网络,主要包括了 4 种 actor 与 lane 之间的交互,基于 LaneGCN 和交互网络实现了多模轨迹输出。一 个完整的行为预测模型包含 4 个部分:

- 1. ActorNet, 主要由 1D 卷积神经网络(Convolutional Neural Network, CNN) 和特征金字塔网络(Feature Pyramid Networks, FPN)构建主要用于提取障碍物历史特征。
- 2. MapNet,从矢量化地图数据中学习结构化地图表示,从矢量化地图数据中 构建道路图像,构建完成LaneSegment的稀疏图,并构建一个LaneGCN提取LaneSegment特征。
- 3. FusionNet,将来自 ActorNet 和 MapNet 的 actor 与 lane 节点信息进行信息融合,4个串行的 attention block 建模交互。首先是 actor2lane,将实时的交通流传递给 lane node;然后 lane2lane 用一个 laneGCN 将信息从一个 node 传递给另一个 node 更新 lane node 特征;然后 lane2actor 将更新的地图信息传递给 actor;最后是 actor2actor 考虑障碍物之间的交互。
- 4. Header 解码输出多模轨迹

而一个典型的 laneGCN 模型如图 3.9所示。基于该模型和 actor-map 的交互,能够实现预测准确的多模轨迹。

3.3.3 VerifAI 下 Scenic 变异扩增脚本测试方法

Fremont 等人 [99] 的工作中同样使用了基于 Scenic 和 VerifAI 的方法来测试 自动驾驶系统,使用来自人工编写的一个特定 Scenic 脚本构建的环境、轨道测 试数据以及他们定义的用于评估预测数据并分析 AI 模型安全操作的数学指标。 本研究提出的用于验证变异 Scenic 脚本的方法参照了这一思路,基于 LaneGCN 模型,本研究完整地设计和实现了在 VerifAI 组件支持下的变异动态场景的轨迹 预测测试方法。



图 3.9: LaneGCN[1] 网络结构

与 Fremont 的方法不同,本研究中的场景对象来自多个变异 Scenic 动态场 景脚本,并进行了多次测试以分析不同类型变异的误差;没有使用仿真轨迹数 据作为 ground truth,而是借助特定自动驾驶行为预测模型获得轨迹预测结果, 将模拟器中的历史时间戳中的交通数据作为输入,将模型当前轨迹作为 ground truth,以达到分析和测试目标行为预测模型的目的。

本研究的方法使用来自外部提供的三个输入: A 设置的多目标评估指标 B 行为预测模型 LaneGCN C 变异 Scenic 样本构建的一组仿真场景,提供多种类型数据来测试模型。测试主要方法流程归纳如下:

- 1. 训练目标行为检测模型,配置执行模型验证的参数,作为 VerifAI 组件的 验证模型。
- 使用 4.3.1 节中重新处理的变异脚本,运行 Carla 模拟器生成的多组具体 场景,并在每次渲染过程中采集模拟器中的各类型数据,采集方法使用 Scenic 提供的 VerifAI 支持库实现。
- 针对采集的传感器数据,包括 RGB 数据、景深相机数据激光雷达数据以及重要的 Scenic 中交通参与对象的轨迹信息(渲染过程中在多个时间戳的位置信息和行驶方向)进行数据处理:

(1)检查采样器获取的传感器信息,去除其中非 RGB 摄像头、景深摄像头、语义分割摄像头和激光雷达信息,保存采样器中获得的轨迹信息。

(2)处理历史轨迹信息,从采集到的数据中将一部分历史轨迹信息提取出 来作为待定行为预测模型的输入,包括不同时间戳下的交通对象类型、对 应时间戳的二维位置信息数据和传感器采集的数据。

定义计算预测轨迹和实际轨迹差异的定量度量指标,在实时测试过程中进行计算。利用该指标, VerifAI将会判断给定测试场景下模型输出是否存在不安全行为。

3.3 基于 VerifAI 的车辆行为预测模型测试方法

5. 可以使用 VerifAI 通过采样搜索具体的故障场景,并将 Carla 记录的识别的 故障具体场景信息记录在本地。

本研究首先设计了动态的 Carla 全类型数据采集,然后同时动态执行验证流程,具体算法设计见算法 2。

VerifAI 通过以下的方式实现 Scenic 模型支持,提取一个 Scenic 场景的抽象特征空间,这些空间由场景的语义特征,例如构建的交通实体位置、定义的特征分布组成、交通参与者的行为等。在多目标评估指标的评估指标上,可以为不同级别的指标设定优先级,并使用监视器 monitor 进行具体的计算评估。论文设计的算法 2中,第 26 行是关键的设计和计算用于预测轨迹评估的指标。待测试模型在轨迹跟踪任务中输出为多个目标的轨迹预测值,相对的,在 VerifAI 中有很多指标可以用于验证评估类似的系统,VerifAI 使用度量时态逻辑(metric temporal logic, MTL)的稳健语义来计算安全指标的定量满足值 ρ ,该值指示满足的程度: $\rho > 0$ 意味着满足 φ , $\rho < 0$ 的较大值意味着模型预测与 ground_truth 产生了较大偏差。

使用的第一个评估指标为最终位移误差(Final Displacement Error, FDE),即在最后一个时步内预测到的轨迹点与 ground_truth 给出的轨迹点之间的欧氏距离:

$$FDE(x, y) = \sqrt{(x_n - \hat{x}_n)^2 + (y_n - \hat{y}_n)^2}$$
(3.6)

第二个指标为平均位移误差(Average Displacement Error, ADE),指在 n个时步内预测得到的轨迹点与 ground truth 中轨迹点之间的均一化欧式距离:

$$ADE(x, y) = \frac{1}{n} \sum_{t=1}^{n} \sqrt{(x_t - \hat{x}_t)^2 + (y_t - \hat{y}_t)^2}$$
(3.7)

模型可能输出严重违反 groundtruth 的预测,而该预测值实际上是合理的,因此,本论文引入了另外 2 种行为预测评估指标以减少此情况带来的误差,这些指标建立在对于最终位移误差和平均位移误差的修改:

最小位移误差 (minFDE), 定义为在某时步的样本, 其最小位移误差为所有 N 次预测在此处最可能的预测结果中最终位移误差值的最小值:

$$\min FDE\left(P_{K}^{i}\right) = \min_{(x,y)\in P_{K}^{i}} FDE(x,y)$$
(3.8)

最小平均位移误差(minADE),定义为在某时步的样本,其最小位移误差为所有此处最可能的 K 条预测结果中平均位移误差值最小值:

$$\min ADE\left(P_{K}^{i}\right) = \min_{(x,y)\in P_{K}^{i}} ADE(x,y)$$
(3.9)

考虑到 *minFDE* 和 *minADE* 只能评估最好的一个轨迹,本方法还引入了未中率 (Missing Rate,MR),预测点迹之间欧式距离高于设定的阈值 *e* 的样本在所有轨迹中的比例:

$$MR(P,\varepsilon) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left\{\min FDE\left(P^{i}\right) > \varepsilon\right\}$$
(3.10)

总的来说,本方法希望获得的 minADE、minFDE、MR 都是越小越好。

```
Algorithm 2: Scenic 数据采集与验证算法
1 #all data:scenic 所有类型数据采集参数, final result:VerifAI 获得的数据
    结果
2 verif_samptype ← VerifAI.SAMPLER_TYPE
3 if verif samptype not None then
      sensors params \leftarrow verif samptype
 4
5 sensors params \leftarrow {}
6 foreach scenic file in scenic path do
7
      all_data[sensor_params] ← sensors_params
      foreach data in scenic out do
8
         if data in scenic sensor list then
 9
             all data[sensor params] \leftarrow True
10
      all data[sensor params] \leftarrow False
11
      sampler \leftarrow ScenicSampler.from(scenic path, all data)
12
      START TO SAMPLING
13
      final result \leftarrow sampler.collect()
14
15 #为目标预测模型处理历史轨迹数据, timepoint 按照此时间戳进行分组
16 file \leftarrow file.open(traj path, "w")
17 # 采集所有交通参与者轨迹数据构建测试输入
18 file.writerow(['TIMESTAMP', 'TRACK ID', 'OBJECT TYPE',
  'X', 'Y', 'CITY NAME'])
19
20 foreach timestp in range(timepoint-20, timepoint) do
      foreach obj type, position in enumerate(traj data[timestp]) do
21
         if obj type == '0' then
22
             obj type \leftarrow' AV'
23
         if obj type == num agents -1 then
24
             obj type \leftarrow' AGENT'
25
         obj type = 'ELSE'
26
           file.writerow([timestp,id,obj type,position[0],position[1],city name])
27 file.close()
28 计算轨迹评估指标值
29 CACULATE METRICS
30 print(result)
```

第四章 实验验证与分析

在本章,基于 Scenic 变异脚本代码和构建的变异测试数据,进行了具体的 实验。这些实验将回答 3 个问题:

- **研究问题1**提出的变异方法是否有效,变异获得的 Scenic 样本,能否真正有效 地发现已有自动驾驶模型系统潜在的问题?
- **研究问题 2** 生成的变异数据能否验证测试简单自动驾驶模型系统,能否借助基于规则的变异策略解决测试 AI 模型结果分析难点,发现问题并增强模型?
- 研究问题3生成的变异数据,能否能否有效评估和验证复杂自动驾驶模型系统,借助自动驾驶行业的通用指标,为测试数据生成和模型验证提供有效帮助?

为了回答上述3个问题,验证第三章提出技术方法的有效性,本论文设计 并进行了3项具体实验,从多个可验证的角度,深入分析了本技术在设计上的 合理性,以及应用于自动驾驶、智能驾驶领域感知模型和基于感知能力测试的 复杂系统,进行有效的软件质量保障的可行性。本实验使用的相关软硬件设备 和实验环境见表格4.1。

4.1 实验准备

4.1.1 数据筛选与采样

在实验环节进行了数据筛选,尽管预处理后数据本身质量得到保障,但是 大量生成的数据样本需要进行筛选,否则会产生一系列的问题,例如,多帧连续 渲染的场景图像常常超过需求数量,同时考虑到不同场景的渲染时间存在差异, 生成场景数量也会存在不同,这必会导致获得的图片在给定场景上的分布存在 问题。期望变异的样本要求不同,在缺陷检测阶段是能够检测模型潜在缺陷的 边界用例,而在数据训练阶段,则更多的想要保证样本在多种情况下符合真实 的分布,从而保证模型的泛化能力。在数据筛选时,本技术将首先按照设定的 生成策略,为不同类型的场景生成大规模的渲染数据,这些数据将会因为各种 因素影响(例如 Carla 模拟器和 Scenic 库中存在的 bug、变异图像生成逻辑、图 像 resize 操作等),出现图片像素值异常,生成图片内容明显存在缺陷(包括空

	表 4.1: 实验软硬件环境配置
	软硬件信息
OS 环境	Ubuntu18.04
Python 版本	Python3.8
	Python3.6 (虚拟环境下训练和使用 Yolo-v4)
Scenic 版本	2.0.0
Carla Simulator	0.9.12
Carla API	0.9.12
VerifAI	2.0.0b1
环境管理工具	Anaconda
机器学习框架	pytorch 1.5
cuda	11.4
CPU	intel i9-12900K
显卡	NVIDIA RTX 3080Ti (12G)
内存	32GB DDRM4

图像输出、训练图像集合中不合理的数据分布)等现象,导致缺陷样本不平衡、数据特征缺失问题。在具体的研究和实验中,本论文使用了以下一些关键的数据和数据集处理策略:

生成阈值与数据缺失处理 在很多后续的研究和实验中,为基于独立脚本生成数 据设定了阈值,以保障以单一场景脚本为基础的渲染场景数据数量保持稳 定,针对变异后脚本,同时将限制其中一些重要特征出现的频率,并清除 不满足要求的缺陷或冗余数据。在一定的阈值控制下,对于缺失样本,进 行缺失值处理。在本文获得的多种数据中,存在一些不含缺失值的变量,但 是在一些传感参数中,存在一些含有缺失值的不完全变量,例如通过删除 道路实体类的变异脚本,模拟器渲染场景导出的数据中部分样本缺少了速 度属性,或者在一些模拟轨迹采集数据中缺少对应的历史信息数据。在多 种情形下,考虑数据缺失类型,进行有效处理。对于完全随机丢失(MCAR, Missing Completely at Random)即数据的缺失与任何假设值和变量之间不 存在依赖关系,同时也不会影响数据本身的偏差程度,属于完全随机出现 的现象。这类情况下根据其出现情况删除缺失值的数据。

对于随机丢失 (MAR, Missing at Random),除了观测数据之外,数据本身 不会对数据数据丢失概率产生影响。数据的缺失并非完全随机的,而是依

赖于部分完全变量,这种情况下,根据其出现情况删除缺失值的数据,使 用数据加权方法,对于不同数据个案分配不同的权重(基于 logistic、probit 等回归算法得到)以减小偏差。

对于非随机丢失 (not missing at random, NMAR)数据的缺失与不完全变量 自身的取值有关。可以使用一些均值插补法,例如同类均值插补,极大似 然估计 (Max Likelihood),多重插补 (Multiple Imputation, MI)等方法尝试 解决该问题。在本文中,遭遇到的缺失值问题较为简单,同时数据样本属 性较少,对于出现的问题,一般使用的是均值插补和同类均值插补法。均 值插补依据缺失值类型进行划分:如果缺失值是定距型的,使用已有值的 平均数进行插补;如果缺失值是非定距型的,将目标属性所有数据的众数 作为缺失补齐值。同类均值插补也是一种单值插补,用层次聚类模型先进 行缺失值类型预测,再以该类型的均值插补。

均匀分布采样 在一些需要生成数据满足真实合理的分布条件下,使用了均匀分 布采样的策略,例如,针对变异的天气属性进行场景变异时,每次渲染时, 从设定需要加入的变异天气条件集合中进行均匀采样,以保障最后获得数 据在这一特征上的质量,避免出现后续构建数据集时缺少特定天气条件样 本或者极端不均衡问题,影响模型性能。均匀分布的概率密度函数为

$$f(x) = \frac{1}{b-a}, a < x < b$$
(4.1)

$$f(x) = 0, else \tag{4.2}$$

a,b为边界,一阶矩为

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx = \int_{a}^{b} \frac{x}{b-a} dx = \frac{a+b}{2}$$
(4.3)

基于此分布,可以推导获得离散型均匀分布,若随机变量有 n 个不同值, 且这 n 个值概率相同,则称之为离散均匀分布,这种分布通常代表不同情 况发生的机会是不确定的,但是发生的可能性是相同的,对应于上述例子 中提到的生成交通场景的天气特征。

4.1.2 数据预处理

本文在后续涉及 Yolo-v4 和 LaneGCN 两个深度学习模型的数据处理中使用 了特征标准化。本研究获得的图像数据进过了图像标准化和图像白化的预处理。

图像标准化是在进行图像相关神经网络操作时一种典型的预处理方法,图像标准化指去除像素点均值以实现中心化处理,在概率分布上,这种方法使得

数据符合数据分布规律,对应训练模型具有更好的泛化能力。对于通过模拟器 渲染获得的数据,考虑到模拟器和算法输出图像方式,可能会造成获得 Scenic 场景对应图片文件本身存在差异,为了保障模型获得生成图像中不受外界影响 的恒定信息,需要进行白化处理。简单来讲,图像标准化就是使得数据的每一个 维度具有零均值和单位方差,将图片分布变化为(0,1)的高斯分布,除此之外, 大小、图像通道数据与原图像保持一致。图像标准化的处理如下:

$$image_standardization = \frac{X - \mu}{adjusted_std}$$
(4.4)

$$adjusted_std = \max\left(\sigma, \frac{1.0}{\sqrt{N}}\right)$$
 (4.5)

公式 3.1 中, μ 是当前图片的像素平均值, X 为图像矩阵。公式 3.2 中, σ 表示 标准方差, N 表示图像 X 的像素数量。

图像白化也是机器学习里面常用的一种规范化数据分布的方法,可以降低 输入的冗余性,降低特征之间的相关性,使得所有的特征具有相同的方差。这对 于后续输入神经网络模型之中是由很大益处的。白化方法分为 PCA (Principal Component Analysis,主成分分析)白化和 ZCA 白化,其中 PCA 白化本质上是 通过线性变换将原始数据变换为一组各维度线性无关的表示,主要包含一下几 个步骤:

- 将原始数据图像数据按列组成 n 行 m 列矩阵 X
- 将矩阵的每一行进行零均值化,减去所在行的均值
- 求出协方差矩阵

$$C = \frac{1}{m}XX^T \tag{4.6}$$

- 求出协方差矩阵的特征值及其对应的特征向量
- 将特征向量按照对应特征值大小从上到下按行排列成矩阵, 取前 k 行组成 矩阵 P
- *Y* = *PX* 即为降维至 k 维后的数据

ZCA 白化与 PCA 白化类似, 定义白化结果

$$x_{ZCAwhite} = U x_{PCAwhite} \tag{4.7}$$

4.2 实验 1: 基于变异的道路目标识别模型缺陷检测

其中 U 为 PCA 白化中的特征向量矩阵,这是基于 $UU^T = U^T U = I$ 的原理,得 到的协方差矩阵仍然是单位矩阵。

经过图像标准化和白化,可以将渲染得到的图像作为训练或者测试样本。 图4.1显示了一个经过白化后的 Scenic 脚本渲染输出的图像数据,左图为进行图 像白化和标准化处理之前获得的一张场景仿真图像,右图为处理后图像数据,可 以看出,预处理前后的图像存在明显的差别。



图 4.1: 图像白化处理

4.2 实验 1: 基于变异的道路目标识别模型缺陷检测

研究问题1主要探究提出的Scenic变异方法是否有效,为了解决该问题,设 计实验1,该实验遵循一个简单地设计思路:大规模生成变异样本,测试自动驾 驶目标检测模型。实验选择了基于一个种子场景脚本的Scenic 代码所变异的新 的场景样本,这些样本包括来自3种类型的种子类型的不同变异模式,利用这 些变异后的脚本随机渲染多种变异类型的交通场景图像。接着,分别将这些变 异样本进行模拟器仿真,生成了900张不同的场景图片。将这些图像使用预训 练的目标检测模型Yolo-v4进行推理,并审查图像识别任务中发现的目标识别错 误、识别目标缺失问题,以及对模型预测结果中重要指标置信度分析,判断待测 模型是否存在潜在性能缺陷。

4.2.1 数据说明

在本实验中使用了 Udacity 自动驾驶数据集(Udacity Self-Driving Dataset), 这是目前自动驾驶领域几个重要的数据集之一。该数据集是 2016 年 Udacity 为 自动驾驶算法比赛准备的,其中对于交通场景中的连续视频的图片进行了标注, 一共分为两个子数据集4.2,其中数据集1包含汽车(Car)、行人(Pedestrian)、 卡车(Truck) 三种类别,共计9420 张图片; 而数据集2在其中的基础上包含了 更多的类别,增加了骑自行车的人(biker)、交通灯(traffic light,带有属性 Red、 RedLeft、Green、Yellow、YellowLeft)类别,共计15000 张交通场景图片。本实 验中,考虑到 Scenic 脚本生成的图像中交通灯与自行车出现频率较高,选择了 更为复杂的数据集 2 作为初始训练的数据集。

表 4.2: Udacity 数据集类别说明				
Udacity 自动驾驶数据集				
Dataset1 Dataset2				
Car	Car			
Truck	Truck			
Pedestrian	Pedestrian			
	traffic light Red			
traffic light RedLeft				
traffic light Green				
traffic light Yellow				
	traffic light YellowLeft			

对于 Udacity 自动驾驶数据集进行了预处理,编写了对应的处理脚本。处理 流程如下:

- 首先检查了初始的15000张图片对应的标注信息,发现该其中存在少量缺 乏对应标注信息的情况,因此在处理算法中剔除,并删除数据集中对应的 不含标注信息的图像文件。
- 2. 去除标注信息中包含的是否遮挡参数,同时对于标注信息中的目标类别做 了调整,由于使用的 Carla 模拟器中对于交通指示灯的模型支持不够充分, 无法保证生成图像数据中交通指示灯质量,因此将 Udacity 自动驾驶数据 集标注信息中的交通指示灯类别合并成"traffic light 交通指示灯"类别。
- 3. 生成 VOC 格式的标注文件再转化为 Yolo-v4 所需格式。这一步使用官方提供的标注信息转化工具实现。
- 参照本章第一节的数据预处理方法,将数据集中的图像进行的图像标准化 和图像白化操作。

整体处理的算法见算法3:

使用标注转化工具处理完成后,生成 xml 格式保存的标注信息文件,以及用于 Yolo-v4 模型使用的标准格式:实验中,将 Udacity 自动驾驶数据集中的数据

```
Algorithm 3: Udacity 自动驾驶数据集处理算法
1 #label dir 标注目录, img dir 图片目录
2 label list ← []
3 \text{ img list} \leftarrow []
4 foreach parent, dir, filenames in label dir do
       foreach label name in filenames do
 5
           label name \leftarrow label name[: -4]
 6
           label lsit.append(label name)
 7
8 foreach parent, dir, filenames in img dir do
       foreach img name in filenames do
 9
           img name \leftarrow img name[: -4]
10
           img.append(img name)
11
12 label set \leftarrow set(label list)
13 img set \leftarrow set(img list))
14 com \leftarrow img set.difference(label set)
15 foreach item in com do
       os,path.exists(item)
16
       remove(item)
17
18 file \leftarrow open(uda \ label \ path, 'r').readlines()
19 outputfile \leftarrow open(labels.txt,'w')
20 foreach line in file do
       line[4] in ['1','0'] outputfile.write(line.replace('1',").replace('0',"))
21
       line[6] in ['RedLeft', 'Red', 'Yellow', 'YellowLeft', 'Green']
       outputfile.write(line.replace(line[6],"))
22 candidateList.reorderByIRValue();
```

按照训练集 9: 测试集 1 的比例进行划分。训练参数中,主要训练参数 batch_size 设置为 8,subdivision 设置为 1, max_steps 设置为 12000, steps 为 [9600,10800], 将输入图像大小调整为 608*608,模型学习率设置为 0.0013。在生成预测结果中 绘制目标检测框的置信度阈值设置为 0.6。

4.2.2 实验设计

实验选择了基于一个种子场景脚本的 Scenic 代码(base_seed3.scenic)所变 异的新的场景样本,这些样本包括来自4.1节中定义18种子类型15种随机不同 变异策略(去掉了3种指定类型变异策略),利用这些变异后的脚本随机渲染多 种变异类型的交通场景图像。接着,分别将这些变异样本进行模拟器仿真,生 成了900 张不同的场景图片(原始脚本和每种子类型变异脚本各自生成了50 张, 天气与时间类变异较多,额外随机选择其他几种可变异内容构建复杂脚本生成 150 张)。将这些图像使用目标检测模型 Yolo-v4 进行分析,并审查图像识别任务 中发现的目标识别错误、识别目标缺失问题,以及对模型预测结果中重要指标 置信度分析。实验1的目的在于证明变异方法下生成的变异图片能够发现模型 潜在的问题并分析问题类型与原因,证明其鲁棒性和安全性上存在隐患。采用 随机策略生成的图像没有进行标注,因此没有使用具体的精准度和召回率。

使用的评判指标之中,目标误检数即为预测结果中成功识别出了置信度阈 值以上的对象,但是错误地进行了分类的图片数量,实际上对应着目标检测中 FP 的概念,见图4.2;



图 4.2: 随机变异下误检问题

目标漏检数预测结果中对于应当被划分为"人、自动车骑行者、汽车、卡车、 交通指示灯"类的对象出现了遗漏的数量,这里的统计类似目标识别中的 TP 用 例中的未检测的一部分,(随机生成图像本身没有 groundtruth,因此无法区分是完 全未检测到对象还是检测到但是小于阈值造成的未识别),见图4.3;

平均置信度达到置信度阈值的检测图像其所有识别对象的平均置信度; 最低置信度指检测目标中置信度最低值。



图 4.3: 随机变异下漏检问题

4.2.3 结果分析

进行实验的具体结果如表格4.3所示。分析对应的指标统计结果,可以看到随机生成的 900 个测试图像在训练后模型的输出中,出现目标误减的样本数为 28 个,占比为 3.1%,出现目标漏检的样本数为 159 个,占比为 17.7%,在置信度 阈值设置为 0.6 的情况下,变异样本中的最低置信度为 0.6196。

从目标误检数来看,绝大多数情况下出现目标误检这一类型问题的情况较 少,在天气变异中变异场景为夜晚时,目标检测误检数量相对较高,约10%的 样本出现了该问题,组合变异中,误检问题在总计150个样本中出现了11次, 其他变异情况下较少,几乎均为极端情况下的个例。

从目标漏检数来看,该类型问题发生频率明显高于误检问题,所有随机变 异类型下均出现了该问题,一共发现了159个漏检样本,其中对于场景时间变异 至夜晚,场景天气变异至雨天,和向量偏移变异中漏检比例最高,分别为52%, 28%和30%。分析认为,这3种类型下生成的变异图像样本中变异后部分对象 的像素数据受到了较大影响,如夜晚和雨天条件下,像素值数据本身受到了直 接影响,影响了模型对于对象边框和置信度的判断,而为代表位置或相对偏移 的向量进行修改变异的操作,也容易导致目标对象生成图像出现问题(新的位 置过远导致像素变小,或者过近超过图像边界导致失去部分可识别内容)。

添加天气变异中的雨天变异、时间变异的夜晚变异、以及代码中数值与关 系变异对于目标识别的平均置信度影响最大,这从某个角度说明这几类变异构 成的数据整体更加难以识别。同时绝大多数的样本最低置信度都接近了预设的 置信度阈值 0.6,这表明了通过变异产生的样本中,基本都成功构造出了达到和 接近模型识别预测下限的极端场景用例,这对于模型的鲁棒性与可靠性验证具 有帮助。

归纳以上结果和相关分析,得出以下结论:

4.2 实验 1: 基于变异的道路目标识别模型缺陷检测

可变异内容	变异方法	目标误检数	目标漏检数	平均置信度(%)	最低置信度(%)
	norom mon 转换为陈扣 5 관地图之	2	6	01.47	71.02
地图	param map 省拱万随机 5 东地图之	2	0	91.47	/1.95
天气	指定变异从 Rain 中的所有天气场	2	17	81.33	67.13
	景中随机均匀采样				
	指定变异从 Clear 中的所有大气场 暑中随机协切平祥	0	3	91.39	82.81
	泉中随机场习不伴 指定变异从 Wet 中的所有天气场景	1	9	87.13	73.33
	中随机均匀采样				
	指定变异从 Cloudy 中的所有天气	0	9	92.36	61.96
	场景中随机均匀采样				
时间	指空恋县从 Night 由的所有于与扬	5	26	77 23	65.92
HĴ [H]	最中随机均匀采样	5	20	11.25	03.72
	指定变异从 Noon 中的所有天气场	0	7	93.57	83.24
	景中随机均匀采样				
	指定变异从 Sunset 中的所有天气场	1	5	89.44	76.20
	景中随机均匀米样				
场景	修改对象获得的 manuever 中驾驶	0	5	93.84	82.67
	类型,随机替换为 LEFT_TURN,				
	RIGHT_TURN, STRAIGHT				
	修改交通路口类型,3way 4way 互换	0	3	96.21	63.59
对象类型	修改同类型 blueprint 为随机模型	2	9	94.42	64.84
~~~~~~	移除目标对象	0	0	92.39	61.96
	修改为可运动其他类型对象,从	1	5	87.93	79.08
	Car, Truck, Bycicle 采样				
数值与关系	标量值, deg 添加偏移取模 180. 距	1	3	77 44	73.11
	离添加偏移量	-	-	,,,,,,,	
	向量值,位置添加偏移	2	15	83.90	73.77
复杂变异	使用天气外的对象、数值和场景的	11	37	86.69	68.28
	补充组合变异				
总计		28	159		61.96

表 4.3: 变异结果

结论 1:利用本研究的变异策略构建的变异 Scenic 脚本,生成的无标注 信息仿真图像能够有效的发现目标识别模型潜在的问题,在小规模的测 试与人工检查结果的条件下,可以帮助发现模型潜在的鲁棒性与准确性 问题。对于结果的统计表明,时间变异中夜晚变异、天气变异中雨天变 异生成的交通场景下模型漏检现象相对严重,数值与关系变异中的标量 值与向量的变异改变了生成图像中对象的相对关系,影响原始模型的预 测可信程度。

# 4.3 实验 2: 基于规则的道路目标识别模型测试与增强

研究问题 2 在确定了生成的脚本有效性后,进一步思考了在规模较大的变 异样本条件下,从评估不同变异方法生成的变异样本到实现模型测试到模型增 强的完整测试流程。人工审查存在成本和效率上的限制,依赖人工的测试评估 是不可靠的,同时很多场景下,难以直接给出模型结果的好坏。随机化的变异 脚本和图像生成在面对在自动驾驶模型这种 AI 驱动的模型时,测试应当具有何 种输出、不同输出中哪些表明模型预测存在问题难以界定。因此,确定用于评 估样本和模型的指标,解决自动驾驶模型的测试 Oracle 问题,是保障本文提出 方法有效性和实用性的重要条件。设计了实验 2,针对本文提出的基于规则的 的变异模式,首先重新训练了用于测试扩增变异数据的目标检测模型,该模型 在实验 1 使用的模型基础上进行了再训练,以学习 Carla 生成的图像特征,将 经过预训练的模型作为待测模型,在构建了种子场景-基于规则的变异场景的对 应关系,并使用不同规则下的评估方法,对变异前后脚本构建的图像进行模型 验证,检测待测试模型潜在的不足,并针对性地进行再训练以检测模型增强能力。

#### 4.3.1 数据准备

实验首先准备用于重新训练 Yolo-v4 模型的数据集,从 Udacity 数据集中,随机抽取了 500 张带有标注的随机样本,接着,从实验 1 的 Carla 渲染的 900 张数据中,选择了平均置信度最高的 500 张图片,使用 labelImg 图像标注工具进行 了图像标注,最终构建了包含 1000 张图像的训练数据集。重新对于 Yolo-v4 模型 进行迁移学习,新构建数据集中的数据按照训练集 9:测试集 1 的比例进行划分。训练参数中,主要训练参数 batch_size 设置为 8,subdivision 设置为 1, max_steps 设置为 12000, steps 为 [9600,10800],将输入图像大小调整为 608*608,模型学习 率设置为 0.0013,在生成预测结果中绘制目标检测框的置信度阈值设置为 0.6。

该重新训练的模型将使用新的变异扩增数据进行测试,这些数据以其构建的基础 Scenic 脚本为划分,并且实验为原始 Scenic 脚本和变异后 Scenic 脚本在 仿真引擎中获得的图像建立关系映射,形成用于测试的成对测试样本,其中变 异中使用的规则为 3.2 章中设计的 8 种规则。处理好的测试样本如下:

$$T_{X,X_{mut}}: X_{mut} = R(X) \tag{4.8}$$

X 为原始 Scenic 脚本运行仿真获得的图像集, X_{mut} 为某特定变异规则 R 下变异 样本获得的图像集。

# 4.3.2 实验设计

总体上,实验2基于预先设定的8种变异规则,分别使用变异 Scenic 脚本在 Carla 模拟器下生成新的图像场景图像作为预训练目标检测模型的测试输入,根 据变异规则中的约束关系分析同一规则变异前后样本的模型输出间关系,统计 违反变异规则的不一致情况。在待测模型预测结果中利用约束完成对变异测试 样本的评估之后,将生成的符合约束的变异样本划分为2类,良好的没有违背规 则的变异样本,称为为一致样本,以及在模型预测中出现明显违背规则情况的 样本,称为非一致样本。本研究的实验基于一定的假设,一致样本其测试结果表 明对应的变异 Scenic 脚本具有一定的合理性,反应了模型在特定类型数据上的 泛化能力。对于出现明显违背约束的变异样本,通过人工检查变异后 Scenic 脚 本的方式,分析是否是因为 Scenic 代码本身在生成过程中出现的质量问题导致 了违反约束的情况,将没有明显缺陷的脚本称为非一致性脚本,构建的图像组 成数据集 Dataset incon。非一致性样本与变异规则中定义的变异前后样本间的 约束关系相关,表明模型在该类型的目标检测任务中存在偏差或其他缺陷,帮 助测试目标模型。同时,实验分析了缺陷产生的原因,可能正是因为在训练样本 集中该类型的数据样本不足,从而导致的泛化能力不足,因此,使用非一致性样 本的脚本生成新的增强测试图像,对于目标检测模型进行再训练,继续分析检 测结果中平均准确度和召回率等关键指标,以验证组成的数据集是否具有在模 型训练验证和测试上的价值。

#### 4.3.3 结果分析

使用重新组建的数据训练的 Yolo 模型在测试集上的性能如图4.4所示,在模型训练和测试过程中采集了全部迭代过程中的结果。可以看出,新训练的模型在经过 20k 次的迭代下,各种条件下的平均准确度和回归率等各项性能参数趋于收敛,且基本具备了实际用于道路目标识别的效果。本研究进一步对于具体的指标进行了统计和分析,去除了训练过程中出现的极端和不合理的指标结果,最终得到了表格4.5中的详细结果。位于该表格第一行,即初始训练后模型的各项平均参数中,平均模型准确度 AP 达到了约 0.51,在交并比 IoU 分别使用 0.5和 0.75的情况下,在测试集上的平均准确度 AP50, AP75 两个指标中分别达到了 0.68和 0.49,在不同像素的目标检测中,平均准确度具有较大的差异,其中小像素和中等大小像素结果 AP_small 和 AP_medium 中的结果分别为 0.35和 0.32,在大像素目标中,模型表现出较高的识别水平,AP_large 性能达到了 0.62。准确率分析了在目标检测任务中预训练模型对于已识别为正样本目标的分类性能。在召回率上,分别给定 1次、10次和 100次的检测中最大召回率 AR1、AR10和



4.3 实验 2: 基于规则的道路目标识别模型测试与增强

图 4.4: 预训练模型性能变化

AR100 分别达到了 0.43、0.59 和 0.61。在不同像素大小的检测任务中, 召回率 AR_small、AR_medium 和 AR_large 分别到达了 0.46、0.51 和 0.83。召回率分析 了在目标检测任务中预训练模型对于所有为正样本目标的检测性能,上述平均 准确度和召回率的实验结果表明已训练模型具备了初步的泛化性能。

接着,本文进行了基于规则的变异数据样本生成与测试任务,使用上述已 具备一定效果的训练后 Yolo-v4 模型进行测试。使用设计的 8 种变异规则,分别 生成了 120 个变异图像样本与种子场景生成图像组合成测试对(其中规则 1、2、 4 涉及了 3 种参数配置),并基于对应规则的预期输出的约束关系检查在训练模 型上的具体置信度和目标检测数量结果,统计了出现与变异样本输出不满足约 束的不一致样本。该测试统计结果见表格4.4。

对于实验结果进行分析,首先,利用基于规则变异的数据生成和模型测试 方法可以有效判断模型输出质量,在所有生成的测试样本中,共计发现了198例 违反规则定义约束的样本,表明利用输出的一致性关系可以检查模型在特定类 型交通场景下检测的安全性和准确性,可以在无标注的情况下执行大规模的测 试分析,一定程度上解决变异脚本仿真图像测试的 Oracle 问题。然后,不同的 变异规则生成的测试脚本,其仿真获得的图像与原图像在模型测试中出现不一 致现象的比例存在差异。在所有的变异规则中,规则2在场景中添加行人,规则 4 雨天变异中的大雨天气变异,规则6 阳光角度变异以及规则7 距离标量偏移

4.3 实验 2: 基于规则的道路目标识别模型测试与增强

<b>我</b> 市.0年又开观州于	协风大型印水	
变异内容	不一致输出	规则约束违反比
	数量	例
添加汽车/卡车对象t个,与	(t=1)7	(t=1)5.83%
默认车辆距离低于可见距	(t=2)6	(t=2)5.00%
<b>这</b>	(t=3)11	(t=3)9.17%
添加行人对象 t 个, 与默认	(t=1)26	(t=1)21.67%
车辆距离低于可见距离	(t=2)22	(t=2)18.33%
	(t=3)29	(t=3)24.17%
变更对象模型	5	4.17%
雨天变异	(小雨)4	(小雨)3.33%
	(中雨)3	(中雨)2.50%
	(大雨)19	(大雨)15.83%
道路积水变异	4	3.33%
阳光角度变异	35	29.17%
距离标量 dis 添加偏移 b	16	13.30%
向量 vec 添加偏移 b	11	9.17%
	<ul> <li>变异内容</li> <li>添加汽车/卡车对象t个,与</li> <li>默认车辆距离低于可见距离</li> <li>添加行人对象t个,与默认</li> <li>车辆距离低于可见距离</li> <li>变更对象模型</li> <li>雨天变异</li> <li>道路积水变异</li> <li>阳光角度变异</li> <li>距离标量 dis 添加偏移 b</li> <li>向量 vec 添加偏移 b</li> </ul>	変异内容       不一致输出数量         添加汽车/卡车对象t个,与       (t=1)7         默认车辆距离低于可见距       (t=2)6         离       (t=3)11         添加行人对象t个,与默认       (t=1)26         车辆距离低于可见距离       (t=2)22         变更对象模型       5         雨天变异       (小雨)4         化中雨)3       (大雨)19         道路积水变异       4         阳光角度变异       35         距离标量 dis 添加偏移 b       16         向量 vec 添加偏移 b       11

表 4.4: 8 种变异规则下测试实验结果

的变异规则下,变异样本出现的不一致输出最多,10%以上的样本出现该问题 (具体的不一致比例从最低13.30%到最高29.17%)。分析这几类变异规则对于 测试数据的影响,对于添加行人的变异规则,由于行人目标在图像中的像素较 小,而预训练模型在初始性能指标中对于小目标检测的平均准确率较小,该变 异规则下模型的目标检测性能较低,容易出现目标检测置信度较低和漏检的问 题,从而导致添加行人后的测试样本输出与原始输出出现违反约束的问题。大 雨变异和阳光角度(白天变异为夜晚)变异下对于图像中已有对象的像素值产 生的较大的污染,对于目标检测模型的目标边界检测和分类产生了相对明显的 影响。位置偏移变异下,与实验1类似,同样会出现对象远离目标车辆造成像 素过小,以及对象过近出现画面遮挡问题,这些问题反应了交通目标检测中与 相对距离相关的极端场景。实验使用出现不一致输出较多的变异规则,生成用 于目标识别模型数据增强的新样本。选择规则2在场景中添加行人,规则4雨 天变异中的大雨天气变异,规则6阳光角度变异以及规则7距离标量偏移4种 规则进行训练数据增强,对于这些变异样本的图像,使用 labelImg 进行了具体 的标注,构建非一致性数据集 Dataset_incon。使用该数据集对于 Yolo-v4 已有数

衣4.5. 1010 侯至住能.AF 指你						
模型	AP	AP50	AP75	AP _{large}	AP _{medium}	AP _{small}
预训练模型	0.51	0.68	0.49	0.62	0.32	0.35
变异数据增强后模型	0.59	0.78	0.57	0.66	0.39	0.42

表 4.5: Yolo 模型性能:AP 指标

	表 4.6:	Yolo 模型	性能:AR	1指标
--	--------	---------	-------	-----

模型	AR1	AR10	AR100	AR _{large}	AR _{medium}	AR _{small}
预训练模型	0.43	0.59	0.61	0.83	0.51	0.46
变异数据增强后模型	0.49	0.71	0.73	0.84	0.62	0.64

据集进行了补充,并重新训练了模型,实验同样采集了模型在学习过程中测试集上的各项准确度和召回率指标,见图4.5。在测试集上,模型收敛后的新训练模型具体性能见表4.5的第二行。结果显示,使用非一致性数据集 Dataset_incon 进行训练集增强后,新的模型性能得到了提升。其中,模型的平均准确度从 0.51 提升到了 0.59,AP50、AP75 两个指标中分别达到了 0.78 和 0.57,小像素和中等大小像素结果 AP_small 和 AP_medium 中的结果分别为 0.42 和 0.39。在召回率上,AR1、AR10 和 AR100 分别达到了 0.49、0.71 和 0.73。在不同像素大小的检测任务中,AR_small、AR_medium 两项提升较为明显,分别提升了 8 和 11 个百分点。而在剩余的指标,大像素目标识别的平均准确率和召回率上,新模型保持了与预训练模型基本一致的性能。

对于该增强结果进行分析和研究,首先,在结果上,模型整体性能尤其是针 对中小像素目标识别的效果得到了明显的提升,新的变异样本中对应变异规则 生成了包含多种小像素目标的有效样本(例如行人、远距离车辆),增强了模型 在该类型小像素目标上检测分类的能力。同时,增强变异数据也使得模型在不 同 IoU 设置的准确度和召回率指标上有一定的提升,4类变异样本为原始数据集 补充了大量不利条件下的交通场景,例如极端天气和昏暗的夜晚场景的环境条 件,表明添加变异数据后数据集质量增强,从提高训练数据规模和数据多样性 的角度帮助模型实现了更好的泛化能力。





图 4.5: 使用非一致性增强数据集后训练模型性能变化

结论 2:结合蜕变测试思想定义 Scenic 脚本变异规则以引导测试数据生成和模型验证的测试方法是有效的,能够用于解决变异样本测试目标检测模型时的 Oracle 问题,检测结果还同时证明了特定变异模式下(大雨天气、无光照夜晚、行人添加、距离偏移)模型的泛化性能相对较弱,更加容易出现预测性能下降的现象。进一步的数据增强实验则表明通过将出现不一致输出较多的测试样本补充进训练集中,再训练的模型在规则对应变异场景下的目标检测准确率、召回率等性能指标上均有所提升,本数据生成与目标识别模型测试方法可以构建数据生成到模型测试到数据增强的正向迭代流程。

# 4.4 实验 3: 基于变异的自动驾驶行为预测模型验证

研究问题 3 则进一步验证了生成的变异场景脚本在更复杂的自动驾驶模型 中协助分析和测试目标模型的能力。除了构建面向纯图像感知的测试数据,设 计了最后一个实验,并证明了使用 Scenic 脚本在运行中提供的类型丰富交通环 境数据和交通参与者数据,可以将测试和验证范围扩大到动态的自动驾驶模型 中。实验基于提出的面向自动驾驶行为预测模型中的验证和测试技术,将目标 模型选定为轨迹预测模型 LanGCN,利用变异样本的渲染场景中获得的历史数 据进行交通环境对象轨迹预测,借助选定的变异后 Scenic 脚本,在 VerifAI 工具 下执行动态的验证与分析任务,从而评估目标行为预测模型验证能力。
#### 4.4.1 数据准备

实验 3 使用 Argoverse[102] 数据集对自动驾驶行为预测模型 LaneGCN 进行 预训练。该数据集是由 Argo AI、卡内基梅隆大学、佐治亚理工学院构建的用于 支持自动驾驶汽车感知算法、三维追踪和行为预测研究的数据集。数据集包括两 个部分: Argoverse 3D Tracking 与 Argoverse Motion Forecasting。该数据集是第一 个包含高精地图的数据集,包含了带有几何形状和语义信息的高精度地图数据, 整体上内部采集了激光雷达数据、RGB 视频数据、车辆前向双目数据、高精地 图数据和 6DOF 数据。LaneGCN 使用其中的行为预测数据,这些数据由 324557 条有效的车辆驾驶轨迹数据组成,每个场景(如图4.6)长 5 秒,用于训练和验 证自动驾驶轨迹预测,而且都包含以 10 Hz 采样的每个跟踪对象的二维鸟瞰图 质心。实验中,LaneGCN 接受来自 VerifAI 采集后处理的输入数据,包含 Scenic 脚本在 Carla 中进行仿真时交通参与者的历史轨迹信息,该信息由 20 个时步的 动态数据组成。LaneGCN 将基于该输入,输出未来一段时步内的预测轨迹信息。



图 4.6: Argoverse 行为预测数据集中的一个动态场景

用于测试的变异的动态场景脚本构建的场景共有 3 个,由动态场景样本 seed6.scenic (3.1 节构造的第一个动态种子场景)变异得到,seed6.scenic 原本描述了一辆目标车辆在 3 路十字路口进行左转的行为。

1. 场景 1 在 seed3 中进行了对象变异中,额外车辆添加的变异方法,添加了

#### 4.4 实验 3: 基于变异的自动驾驶行为预测模型验证

一个基于当前目标车辆的逆向车道行驶汽车,将在路口进行继续直行操作, 目标车辆保障避免发生碰撞。

- 场景2在场景1中进行了场景变异与对象变异,将3路路口变异为4路 交叉路口,变更了车道类型 intersection;仿真构建时路口场景将进行替换, 添加了一个基于当前目标车辆的逆向车道行驶汽车,将在路口进行继续直 行操作,目标车辆保障避免发生碰撞。
- 3. 场景3在场景1中进行了场景变异和对象变异,将3路路口变异为4路交 叉路口,变更了车道类型 intersection;将额外车辆添加的变异方法,添加 了一个基于当前目标车辆的逆向车道行驶汽车,并在路口进行右转操作; 对目标车辆进行了对象变异,变异路口左转操作为直行,需要保障目标车辆避免发生碰撞。



图 4.7: 使用 seed6.scenic 变异生成的 3 种动态场景

#### 4.4.2 实验设计

Indaheng[103] 等人同样使用了 Scenic+LaneGCN 的验证方法,对于构建的 场景进行测试,本研究中基于他们的思想,分析了不同的变异后动态 Scenic 脚本 构建场景下,如何进行 LaneGCN 的有效验证,不同的是本测试场景来源不是手 工构建的场景脚本,而是基于种子场景变异后脚本代码,同时不仅关注 VerifAI 获得的验证结果,还进一步分析了借助自动化变异由同一份 Scenic 场景扩增的 变异场景在测试脚本下的差异。使用 CARLA 模拟器生成实验变异的 3 个具体 场景。从每个模拟中收集各种类型的数据以测试行为预测模型,从而测试目标 LaneGCN 行为预测模型。收集轨迹(即场景中所有交通参与者的时间戳位置和 航向)和传感器数据流,提取两种不同类型的数据:(1) 历史数据,这是输入到 行为预测模型的数据的分段,以及(2)地面真实未来轨迹数据,用于使用提供的 多目标评估度量来评估模型的预测轨迹。

具体而言,实验将变异生成的场景1,场景2,场景3,分别采样和生成具体的场景,并按照不同的预测起始时步划分不同批次。基于一段20个时步的历史数据,将其输入LaneGCN模型,可以预测出未来时步内的多模轨迹信息(图4.8)。在实验设置上,训练LaneGCN使用的batch_size为32,学习率设置为



图 4.8: 模型预测示意图

0.001, predic_size 和 predic_step 分别设置为 30 和 1。变异前后车辆速度较低,因此变异未命中率 MR 阈值 *ε* 设置为 1 米。本文在研究中还发现, Indaheng 等人使用 LaneGCN 进行的 15 个时步预测效果较差,且预测初始时步集中于车辆行驶的前半部分,在较低的车速限制下 20 个时步中车辆历史轨迹接近直行,这对模型的预测产生了不利的影响,本研究认为这导致了其实验中,轨迹预测偏差过大和获得的大多数平均位移误差和最终位移误差较小的问题。本文的实验对此进行了优化,在预测中,仅使用 LaneGCN 预测未来 10 个时步内的数据,同时,选择了不同的预测起始时步 (40,80,120),用于查看车辆在不同行驶时期下的轨迹与 LaneGCN 预测数据的差异,综合判断该轨迹预测模型性能。对于每个变异 Scenic 脚本,分别针对进行了 150 次模拟,最终的指标取平均值。计算获得的最小最终位移误差(minADE),最小平均位移误差(minADE)和未中率(MR)结果见表4.7。

变异场景	minFDE	minADE 预测起始时步 40	MR	minFDE	minADE 预测起始时步 80	MR	minFDE	minADE 预测起始时步 120	MR	
种子场景	0.179	0.042	0.16	0.593	0.072	0.29	0.318	0.027	0.21	
场景1	0.233	0.065	0.33	0.667	0.085	0.33	0.378	0.043	0.33	
场景 2	0.707	0.159	0.41	0.766	0.118	0.49	0.639	0.092	0.56	
场景 3	0.821	0.189	0.40	0.845	0.123	0.51	0.484	0.149	0.42	

表 4.7: VerifAI 验证 LaneGCN 的各指标计算结果

#### 4.4.3 实验3结果分析

所有3个变异场景的150次模拟显示,从不同场景类型上看,在设置起始 预测时间点为相同时步时,变异预测的最小最终位移误差,最小平均位移误差, 以及未命中率均相对于初始种子场景有较大增长,其中起始预测点为40时步时, minFDE从 0.179 增长到 0.821 米, minADE从 0.042 增长到 0.189 米, 考虑到实 验设置的阈值为1米,该结果表现出了明显的预测精度差距。分析认为这是由 于种子样本相对变异场景1,2,3,具有不同复杂度的道路行为和处理方法造成 的,而模型对于复杂的驾驶策略和状况处理方法下的目标轨迹预测能力相对不 足。种子场景图像中目标车辆遵循基本的道路驾驶行为,且没有其他运动中的 交互对象影响,因此不存在需要考虑与其他汽车出现安全距离过近的情况,根 据历史轨迹可以较为准确的预测未来轨迹。变异场景下的各项指标则表明,预训 练的 LaneGCN 模型随着预测场景的复杂程度提升,其预测精度将有所下降,在 3 路路口(种子场景和变异场景1)中,模型的轨迹预测结果相对贴近 Carla 后续 模拟的 ground truth, 但是在 4 路口下, Scenic 生成对应动态场景的状态空间相 对3路口更为庞大,更加容易出现不同类型的交通状态数据作为模型输入,目 标行为预测模型在该种类型上的预测能力出现了明显的降低,表明实验所预训 练的行为预测模型在复杂交通场景下的泛化能力和性能问题。相对与 Indaheng 等人手动编写的脚本,本实验中利用变异自动构建的脚本在复杂性上没有达到 他们编写的最复杂水准,但是同样有效发现了模型潜在的泛化缺陷。

从预测起始时间来看,在预测起始时步为80时,最小最终平均位移误差, 最小最终位移误差,以及未命中率极大提高,出例如在场景2下,目标场景下的 模型预测结果中最小最终位移误差达到了0.63米,最小平均误差也到达了0.49 米,同时模型预测中未命中率也高达0.49,这表明约有一半的预测轨迹误差超过 了设定的阈值。出现该情况的原因在于变异场景中,该部分处于交通交互的中 间时间,常常出现多种需要交通参与对象进行交互和逻辑判断的场景,例如开 始在交通路口实施转弯行为。 结论 3: 基于设计的动态交通场景和变异方法,可以获得用于测试行 为预测模型的多种测试样本,在评估工具 VerifAI 中得到的评估分数表 明了最小最终误差(minFDE),最小平均位移误差(minADE),未命 中率(MR)3个重要评估指标能够成功验证目标检测模型在轨迹检测 任务中的性能。测试结果表明,随着变异模型中交通参与对象的道路 动作行为和不同状况处理逻辑复杂程度的提升,模型预测的效果越差。 同时,在不同时间下交通参与者状态也对模型的鲁棒性造成了不同的 影响,本研究预训练的 LaneGCN 模型在预测包含转向行为轨迹时,出 现了较多的未命中现象,表明在该预测能力上模型存在潜在的性能缺陷。

## 4.5 有效性分析和讨论

本小节分析了本文提出方法和实验中重要的影响因素,以及具体过程中对 于实验设计细节上的补充说明。

Carla 模拟器局限性 在第2章相关方法中介绍了使用模拟仿真引擎进行场景涉 及的优势,同时也分析了该类型的仿真测试带来的缺陷。相对于真实场景 数据,虚拟仿真存在有逼真度相对真实场景不足,同时最大的问题在于构 建场景的丰富性受到模型本身的限制。在本文中使用多种策略,以减少模 拟器的局限性对于技术方法和具体实验的影响。

首先,在场景和交通模型支持上,使用的 Carla 模拟器接口进行交通场景 的渲染时,由于模拟器的限制,Scenic 可以使用的交通对象模型只能为模 拟器自带引擎中模型蓝图的的子集,而即使在最近一个版本中 Scenic 支持 的模型库也并未匹配最新的 Carla 版本。为了保障模拟对象的丰富性,对 于 Scenic 支持的模型做了扩展,添加了 6 种新型号的车辆蓝图和 4 种行人 模型。然后,在具体生成场景过程中,初始 Scenic 代码中对于所有可变异 内容使用了显式代码构造,例如对于天气参数,Scenic 默认在代码中缺省 并使用均匀采样方法进行配置,而本文修改的模板中天气相关代码是明确 定义的,这种方法使得在代码搜索中的可变异内容增多,从而能够生成更 多的不同变异组合的交通场景。最后,在生成具体的交通测试数据中进行 了优化,针对使用变异脚本生成的图像数据,使用图像进行训练时,训练 集采用原始 Udacity 数据集和模拟仿真数据的混合以保障模型的泛化性能, 同时对于目标识别的可检测类型进行修改,例如针对交通指示灯类,由于 模拟器中没有明确指定交通信号灯颜色设置,生成的含信号灯图像中不同 颜色指示灯比例不平衡,因此没有使用 Udacity 数据集中详细的交通指示 灯分类,将多种交通指示灯类型进行了合并,为了保障分类数量,还重新 补充了交通标志的目标类型。

**参数设置** 实验 2 中对于每种定义的变异规则,分别给出用于验证输出的约束关 系表达式,其中的重要阈值,如目标检测偏移 λ 与置信度偏移 ω 设置参照 了实验 1 中随机变异统计的结果。对于出现目标检测偏移(误检和漏检问 题)较多的变异规则,设置了较高的偏移参数(ω 在中雨、大雨和夜晚变 异下分别为 0.2、0.3 和 0.5)。对于模型具体的测试性能结果,实验使用平 滑方法将多个迭代中的图像结果进行了预处理,最终的统计结果中的数值 则来源于模型收敛后的 4 个时间段下(训练到达 20k、40k、60k 和 80k 迭 代)的对应测试集上性能数值取算术平均。

实验3开始之前,首先复现了 Indaheng[103] 中使用人工编写的 Scenic 脚本进行模型预测的实验,在实验过程中,发现了其中存在的一些问题。在他们的实验中,选择的4个预测起始时间局限于模拟驾驶早期,而实际仿真过程中常常没有包含到重要的交通交互(例如转向动作中,原实验中最大起始预测为80 时步,然而实验表明在车辆速度设定条件下往往该时刻还未开始转向行为),这使得很多的测试结果更趋向于车辆执行复杂动作前的直线轨迹预测,同时使用20个时步轨迹数据预测15个未来轨迹的方案中历史轨迹相对信息过少,最终导致实验得到的平均位移误差和最终位移误差没有出现过大偏移。本文中的实验中重新设计了跨度较大的3个时间点(80、100、120 时步),并且仅预测了10个时步的未来轨迹,实验3的结果证明了上述分析,与 Indaheng 等人手动编写脚本对比,类似复杂度的变异生成动态场景下设置覆盖的交通场景更加广泛,尤其是包含复杂交互的时间段(100 时步)中,最小平均位移误差、最小最终位移误差、未命中率 *MR* 出现了明显的提高,发现了更多的模型预测与实际轨迹偏差。

## 4.6 分析总结

本章进行了3个具体实验,以证明本文提出的基于 Scenic 脚本变异的测试 数据生成和模型测试方法的有效性,分别回答变异测试样本能否有效发现自动 驾驶 AI 模型缺陷,如何使用本技术高效测试自动驾驶目标检测模型和如何测试 复杂自动驾驶行为预测模型3个具体问题,基于实验结果归纳得出了3个具体 的实验结论。实验1中通过随机化生成的变异样本测试目标识别模型,发现了 漏检、误检和预测置信度下降的典型问题,表明变异样本确实可以有效发现模 型潜在的问题。实验2参照蜕变测试的思想,定义不同的变异规则有效引导测 试数据生成和测试结果验证,实现了使用无标注变异数据下对于目标检测模型 的有效测试,并利用测试样本中的不一致数据对于训练集进行增强,成功提升 了模型的泛化能力。实验3则进一步地使用生成的更复杂的动态轨迹数据,结 合机器学习分析验证工具,通过分析预测轨迹和模拟器实际轨迹数据之间的量 化性能指标,测试和验证自动驾驶行为预测模型。

# 第五章 总结和展望

## 5.1 总结

场景感知相关的自动驾驶模型质量保障十分重要、测试技术则依赖有效的 高质量交通场景样本数据。当前已有的数据生成和数据增强中,使用领域特定编 程语言进行场景定义减少了直接硬编码变换场景数据带来的可解释性问题,但 是场景代码构建本身缺乏一定的引导,在测试结果验证上也存在问题。本文提 出了一种基于领域特定编程语言 Scenic 的自动驾驶场景数据生成和模型测试技 术,通过对自动驾驶场景构建的代码进行有效变异,生成用于测试自动驾驶感 知和相关模型的高质量测试样本。技术分析了待变异代码中的可变异内容,实 现道路场景图像数据和复杂动态信息数据的生成,基于生成测试数据、设计了 面向自动驾驶道路目标检测模型和自动驾驶轨迹预测模型的测试方法。定义了8 种特殊的变异规则以引导生成变异交通图像样本,并使用规则中变异前后的约 束关系验证目标识别模型输出,将出现不一致输出的样本用于数据集增强。定 义了基于机器学习验证工具 VerifAI 的自动驾驶行为预测模型的测试验证流程, 使用验证工具生成轨迹预测输入,将仿真预测轨迹划分为历史轨迹和后续轨迹, 编写了基于最小最终位移误差(minFDE),最小平均位移误差(minADE)和未 中率(MR)的评估验证指标,通过对比分析模型预测轨迹和真实后续轨迹的偏 差测试分析模型性能。本文进行了3个具体的实验使用实际变异脚本生成的场 景数据在道路目标识别模型 Yolo-v4 和自动驾驶轨迹预测模型 LaneGCN 上进行 了测试验证,证明了该场景构建脚本变异方法和模型测试方法的有效性。

## 5.2 工作展望

本文提出的方法具有一定的应用价值,同时存在很多后续优化的空间和可 扩展的内容。在场景构建脚本代码变异上,可以进一步扩展可变异内容。Scenic 场景构建与使用的 Carla 模拟器息息相关,本文提出的技术中实现了一些重要的 变异内容(天气、环境、位置和交通模型等)的变异,但是很多都不完善,例如 天气条件中定义的变异依赖预定义的不同风格天气类型,这些天气类型则是利 用对于太阳高度、方位角、云密度、降水和雾强度等多种参数的预先硬编码固 化。后续工作中,可以考虑引入更加丰富自由的定制化变异内容支持,扩展可变 异内容组合,以编写包含更加复杂的交通场景的脚本代码。 可以考虑增加模拟器支持,Scenic 支持多种模拟器,且不同模拟器具有不同的优缺点,数据生成受限于 Carla 模拟器本身的限制,通过新增其他模拟器支持,可以生成不同于 Carla 风格的多样化测试场景,获得在场景真实度、多样度或者车动力学仿真上不同水平的测试数据。

扩展测试结果验证和解释方法。使用类似等价关系的测试方法有时不足以 直观解析模型输出结果,通过向变异测试结果引入机器学习等技术,可能会为 偏差输出结果和交通场景实际缺陷行为之间的关系发现与分析提供帮助。提高 变异测试的自动化水平,探索将脚本变异、模型白盒测试与数据增强高效结合 的流程化测试方案,进一步减少人工工作。

# 参考文献

- LIANG M, YANG B, HU R, et al. Learning lane graph representations for motion forecasting[C] // European Conference on Computer Vision. 2020: 541–556.
- [2] MEIRING G A M, MYBURGH H C. A review of intelligent driving style analysis systems and related artificial intelligence algorithms[J]. Sensors, 2015, 15(12): 30653-30682.
- [3] LEVINSON J, ASKELAND J, BECKER J, et al. Towards fully autonomous driving: Systems and algorithms[C]//2011 IEEE intelligent vehicles symposium (IV). 2011: 163–168.
- [4] GRUYER D, MAGNIER V, HAMDI K, et al. Perception, information processing and modeling: Critical stages for autonomous driving applications[J]. Annual Reviews in Control, 2017, 44: 323-341.
- [5] URMSON C, ANHALT J, BAGNELL D, et al. Autonomous driving in urban environments: Boss and the urban challenge[J]. Journal of field Robotics, 2008, 25(8): 425-466.
- [6] BROGGI A, BUZZONI M, DEBATTISTI S, et al. Extensive tests of autonomous driving technologies[J]. IEEE Transactions on Intelligent Transportation Systems, 2013, 14(3): 1403-1415.
- [7] HUANG W, WANG K, LV Y, et al. Autonomous vehicles testing methods review[C] // 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). 2016: 163–168.
- [8] CAMPBELL M, EGERSTEDT M, HOW J P, et al. Autonomous driving in urban environments: approaches, lessons and challenges[J]. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 2010, 368(1928): 4649–4672.
- [9] ZHAO D, LIU Y, ZHANG C, et al. Autonomous driving simulation for unmanned vehicles[C] // 2015 IEEE Winter Conference on Applications of Computer Vision. 2015 : 185–190.

- [10] FREMONT D J, DREOSSI T, GHOSH S, et al. Scenic: a language for scenario specification and scene generation[C] // Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation. 2019: 63 – 78.
- [11] BOCHKOVSKIY A, WANG C-Y, LIAO H-Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.
- [12] BERGER C, RUMPE B. Engineering autonomous driving software[J]. arXiv preprint arXiv:1409.6579, 2014.
- [13] BARABAS I, TODORUŢ A, CORDOŞ N, et al. Current challenges in autonomous driving[C] // IOP conference series: materials science and engineering: Vol 252. 2017: 012096.
- [14] PEI K, CAO Y, YANG J, et al. Deepxplore: Automated whitebox testing of deep learning systems[C] // proceedings of the 26th Symposium on Operating Systems Principles. 2017: 1–18.
- [15] SUN Y, HUANG X, KROENING D, et al. Deepconcolic: testing and debugging deep neural networks[C] // 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion). 2019: 111–114.
- [16] MA L, JUEFEI-XU F, ZHANG F, et al. Deepgauge: Multi-granularity testing criteria for deep learning systems[C] // Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering. 2018: 120–131.
- [17] LI Z, PAN M, ZHANG T, et al. Testing DNN-based Autonomous Driving Systems under Critical Environmental Conditions[C] // International Conference on Machine Learning. 2021: 6471–6482.
- [18] LI Z, MA X, XU C, et al. Structural coverage criteria for neural networks could be misleading[C] // 2019 IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER). 2019: 89–92.
- [19] MARIJAN D, GOTLIEB A, AHUJA M K. Challenges of testing machine learning based systems[C] // 2019 IEEE International Conference On Artificial Intelligence Testing (AITest). 2019: 101–102.

- [20] TIAN Y, PEI K, JANA S, et al. Deeptest: Automated testing of deep-neuralnetwork-driven autonomous cars[C] // Proceedings of the 40th international conference on software engineering. 2018 : 303-314.
- [21] ZHANG M, ZHANG Y, ZHANG L, et al. Deeproad: Gan-based metamorphic testing and input validation framework for autonomous driving systems[C] // 2018 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE). 2018: 132–142.
- [22] CALO A, ARCAINI P, ALI S, et al. Generating avoidable collision scenarios for testing autonomous driving systems[C] // 2020 IEEE 13th International Conference on Software Testing, Validation and Verification (ICST). 2020: 375-386.
- [23] SHAH S, DEY D, LOVETT C, et al. Airsim: High-fidelity visual and physical simulation for autonomous vehicles[C] // Field and service robotics. 2018 : 621 – 635.
- [24] DOSOVITSKIY A, ROS G, CODEVILLA F, et al. CARLA: An open urban driving simulator[C] // Conference on robot learning. 2017: 1–16.
- [25] TETTAMANTI T, SZALAI M, VASS S, et al. Vehicle-in-the-loop test environment for autonomous driving with microscopic traffic simulation[C] // 2018 IEEE International Conference on Vehicular Electronics and Safety (ICVES). 2018: 1-6.
- [26] GAMBI A, MUELLER M, FRASER G. Automatically testing self-driving cars with search-based procedural content generation[C] // Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis. 2019: 318-328.
- [27] HUSSAIN M, ALI N, HONG J-E. DeepGuard: a framework for safeguarding autonomous driving systems from inconsistent behaviour[J]. Automated Software Engineering, 2022, 29(1): 1-32.
- [28] ZHANG C, LIU Y, ZHAO D, et al. Roadview: A traffic scene simulator for autonomous vehicle simulation testing[C] // 17th International IEEE Conference on Intelligent Transportation Systems (ITSC). 2014 : 1160–1165.

- [29] ZOFKA M R, ESSINGER M, FLECK T, et al. The sleepwalker framework: Verification and validation of autonomous vehicles by mixed reality lidar stimulation[C] // 2018 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR). 2018 : 151–157.
- [30] LI W, PAN C, ZHANG R, et al. AADS: Augmented autonomous driving simulation using data-driven algorithms[J]. Science robotics, 2019.
- [31] MU□AT V, FURSA I, NEWMAN P, et al. Multi-weather city: Adverse weather stacking for autonomous driving[C] // Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021 : 2906-2915.
- [32] URICAR M, SISTU G, RASHED H, et al. Let's get dirty: Gan based data augmentation for camera lens soiling detection in autonomous driving[C] // Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021 : 766-775.
- [33] TUNCALI C E, FAINEKOS G, ITO H, et al. Sim-ATAV: Simulation-based adversarial testing framework for autonomous vehicles[C] // Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week). 2018: 283–284.
- [34] DREOSSI T, GHOSH S, SANGIOVANNI-VINCENTELLI A, et al. Systematic testing of convolutional neural networks for autonomous driving[J]. arXiv preprint arXiv:1708.03309, 2017.
- [35] CHEN Y, HU V T, GAVVES E, et al. Pointmixup: Augmentation for point clouds[C] // European Conference on Computer Vision. 2020: 330-345.
- [36] VARGAS RIVERO J R, GERBICH T, BUSCHARDT B, et al. Data Augmentation of Automotive LIDAR Point Clouds under Adverse Weather Situations[J]. Sensors, 2021, 21(13): 4503.
- [37] ZHANG H, GUO Y. Generalization of Reinforcement Learning with Policy-Aware Adversarial Data Augmentation[J]. arXiv preprint arXiv:2106.15587, 2021.

- [38] YU H, SHI W, ALAWIEH M B, et al. Efficient statistical validation of autonomous driving systems[G] // Safe, Autonomous and Intelligent Vehicles. [S.l.]: Springer, 2019: 5-32.
- [39] WONG K, ZHANG Q, LIANG M, et al. Testing the safety of self-driving vehicles by simulating perception and prediction[C] // European Conference on Computer Vision. 2020: 312–329.
- [40] YANG Z, CHAI Y, ANGUELOV D, et al. Surfelgan: Synthesizing realistic sensor data for autonomous driving[C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 11118–11127.
- [41] MAJUMDAR R, MATHUR A, PIRRON M, et al. Paracosm: A test framework for autonomous driving simulations[C] // International Conference on Fundamental Approaches to Software Engineering. 2021: 172–195.
- [42] O'KELLY M, ABBAS H, MANGHARAM R. Computer-aided design for safe autonomous vehicles[C] // 2017 Resilience Week (RWS). 2017: 90–96.
- [43] JO K, KIM J, KIM D, et al. Development of autonomous car—Part II: A case study on the implementation of an autonomous driving system based on distributed architecture[J]. IEEE Transactions on Industrial Electronics, 2015, 62(8): 5119-5132.
- [44] LIN S-C, ZHANG Y, HSU C-H, et al. The architectural implications of autonomous driving: Constraints and acceleration[C] // Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems. 2018 : 751–766.
- [45] WANG J, LIU J, KATO N. Networking and communications in autonomous driving: A survey[J]. IEEE Communications Surveys & Tutorials, 2018, 21(2): 1243-1274.
- [46] BEHERE S, TORNGREN M. A functional architecture for autonomous driving[C] // 2015 First International Workshop on Automotive Software Architecture (WASA). 2015: 3-10.
- [47] ZHANG J, CHO K. Query-efficient imitation learning for end-to-end autonomous driving[J]. arXiv preprint arXiv:1605.06450, 2016.

- [48] DEBOUK R. Safety Strategy for Autonomous Systems Rami Debouk, Ph. D.; Barbara Czerny, Ph. D.; Joseph D' Ambrosio, Ph. D.; Jeffrey Joyce, Ph. D.; Critical Systems Labs. Inc.; Vancouver, British Columbia, Canada[J], .
- [49] STAVENS D, HOFFMANN G, THRUN S. Online Speed Adaptation Using Supervised Learning for High-Speed, Off-Road Autonomous Driving.[C] // IJCAI. 2007: 2218–2224.
- [50] GRIGORESCU S, TRASNEA B, COCIAS T, et al. A survey of deep learning techniques for autonomous driving[J]. Journal of Field Robotics, 2020, 37(3): 362-386.
- [51] SALLAB A E, ABDOU M, PEROT E, et al. Deep reinforcement learning framework for autonomous driving[J]. Electronic Imaging, 2017, 2017(19): 70-76.
- [52] MICHELMORE R, KWIATKOWSKA M, GAL Y. Evaluating uncertainty quantification in end-to-end autonomous driving control[J]. arXiv preprint arXiv:1811.06817, 2018.
- [53] BEHERE S, TÖRNGREN M. A functional reference architecture for autonomous driving[J]. Information and Software Technology, 2016, 73: 136– 150.
- [54] HÖRWICK M, SIEDERSBERGER K-H. Strategy and architecture of a safety concept for fully automatic and autonomous driving assistance systems[C] // 2010 IEEE Intelligent Vehicles Symposium. 2010: 955–960.
- [55] CHEN X, MA H, WAN J, et al. Multi-view 3d object detection network for autonomous driving[C] // Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2017: 1907–1915.
- [56] ARNOLD E, AL-JARRAH O Y, DIANATI M, et al. A survey on 3d object detection methods for autonomous driving applications[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(10): 3782-3795.
- [57] CIBERLIN J, GRBIC R, TESLIĆ N, et al. Object detection and object tracking in front of the vehicle using front view camera[C] // 2019 Zooming Innovation in Consumer Technologies Conference (ZINC). 2019 : 27-32.

- [58] HIRABAYASHI M, KATO S, EDAHIRO M, et al. GPU implementations of object detection using HOG features and deformable models[C] // 2013 IEEE 1st International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA). 2013 : 106-111.
- [59] NAGHAVI S H, AVAZNIA C, TALEBI H. Integrated real-time object detection for self-driving vehicles[C] // 2017 10th Iranian Conference on Machine Vision and Image Processing (MVIP). 2017: 154–158.
- [60] JIAO L, ZHANG F, LIU F, et al. A survey of deep learning-based object detection[J]. IEEE access, 2019, 7: 128837-128868.
- [61] FUJIYOSHI H, HIRAKAWA T, YAMASHITA T. Deep learning-based image recognition for autonomous driving[J]. IATSS research, 2019, 43(4): 244-252.
- [62] LIN T-Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[C] // Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.
- [63] GIRSHICK R. Fast r-cnn[C] // Proceedings of the IEEE international conference on computer vision. 2015 : 1440–1448.
- [64] REN S, HE K, GIRSHICK R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 28.
- [65] LIN T-Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117–2125.
- [66] LAW H, DENG J. Cornernet: Detecting objects as paired keypoints[C]
  // Proceedings of the European conference on computer vision (ECCV). 2018:
  734-750.
- [67] DUAN K, BAI S, XIE L, et al. Centernet: Keypoint triplets for object detection[C] // Proceedings of the IEEE/CVF international conference on computer vision. 2019: 6569-6578.

- [68] TIAN Z, SHEN C, CHEN H, et al. Fcos: Fully convolutional one-stage object detection[C] // Proceedings of the IEEE/CVF international conference on computer vision. 2019: 9627–9636.
- [69] MOZAFFARI S, AL-JARRAH O Y, DIANATI M, et al. Deep learning-based vehicle behavior prediction for autonomous driving applications: A review[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 23(1): 33-47.
- [70] FERNANDO T, DENMAN S, SRIDHARAN S, et al. Deep inverse reinforcement learning for behavior prediction in autonomous driving: Accurate forecasts of vehicle motion[J]. IEEE Signal Processing Magazine, 2020, 38(1): 87–96.
- [71] PHAN-MINH T, GRIGORE E C, BOULTON F A, et al. Covernet: Multimodal behavior prediction using trajectory sets[C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 14074-14083.
- [72] YE M, CAO T, CHEN Q. Tpcn: Temporal point cloud networks for motion forecasting[C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 11318–11327.
- [73] LI J, PAN S, HUANG L, et al. A machine learning based method for customer behavior prediction[J]. Tehnički vjesnik, 2019, 26(6): 1670–1676.
- [74] GENG X, LIANG H, YU B, et al. A scenario-adaptive driving behavior prediction approach to urban autonomous driving[J]. Applied Sciences, 2017, 7(4): 426.
- [75] LUO W, YANG B, URTASUN R. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net[C] // Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2018: 3569–3577.
- [76] SAUER A, SAVINOV N, GEIGER A. Conditional affordance learning for driving in urban environments[C] // Conference on Robot Learning. 2018: 237– 252.
- [77] LIANG M, YANG B, ZENG W, et al. Pnpnet: End-to-end perception and prediction with tracking in the loop[C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 11553-11562.

- [78] GEYER J, KASSAHUN Y, MAHMUDI M, et al. A2d2: Audi autonomous driving dataset[J]. arXiv preprint arXiv:2004.06320, 2020.
- [79] CORDTS M, OMRAN M, RAMOS S, et al. The cityscapes dataset for semantic urban scene understanding[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 3213–3223.
- [80] RAMANISHKA V, CHEN Y-T, MISU T, et al. Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning[C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 7699-7707.
- [81] GEIGER A, LENZ P, STILLER C, et al. Vision meets robotics: The kitti dataset[J]. The International Journal of Robotics Research, 2013, 32(11): 1231– 1237.
- [82] 冯洋, 夏志龙, 郭安, et al. 自动驾驶软件测试技术研究综述 [J]. 中国图象图 形学报, 2021.
- [83] ZABLOCKI É, BEN-YOUNES H, PÉREZ P, et al. Explainability of visionbased autonomous driving systems: Review and challenges[J]. arXiv preprint arXiv:2101.05307, 2021.
- [84] ZHOU J, CHEN H, XIU C. A Simulation Model to Evaluate and Verify Functions of Autonomous Vehicle Based on Simulink®[C] // International Conference on Intelligent Robotics and Applications. 2009: 645-656.
- [85] BENEKOHAL R F, TREITERER J. CARSIM: Car-following model for simulation of traffic in normal and stop-and-go conditions[J]. Transportation research record, 1988, 1194: 99–111.
- [86] XU J, LUO Q, XU K, et al. An automated learning-based procedure for largescale vehicle dynamics modeling on baidu apollo platform[C] // 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2019: 5049-5056.
- [87] ZOFKA M R, KLEMM S, KUHNT F, et al. Testing and validating high level components for automated driving: simulation framework for traffic scenarios[C] // 2016 IEEE intelligent vehicles symposium (IV). 2016 : 144–150.

- [88] SUN P, KRETZSCHMAR H, DOTIWALLA X, et al. Scalability in perception for autonomous driving: Waymo open dataset[C] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 2446– 2454.
- [89] SIPPL C, SCHWAB B, KIELAR P, et al. Distributed real-time traffic simulation for autonomous vehicle testing in urban environments[C] // 2018 21st International Conference on Intelligent Transportation Systems (ITSC). 2018: 2562– 2567.
- [90] GAMBI A, HUYNH T, FRASER G. Generating effective test cases for selfdriving cars from police reports[C] // Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. 2019: 257–267.
- [91] LI L, HUANG W-L, LIU Y, et al. Intelligence testing for autonomous vehicles: A new approach[J]. IEEE Transactions on Intelligent Vehicles, 2016, 1(2): 158– 166.
- [92] VISHNUKUMAR H J, BUTTING B, MÜLLER C, et al. Machine learning and deep neural network—Artificial intelligence core for lab and real-world test and validation for ADAS and autonomous vehicles: AI for efficient and quality test and validation[C] // 2017 Intelligent Systems Conference (IntelliSys). 2017 : 714–721.
- [93] ZHOU H, LI W, KONG Z, et al. Deepbillboard: Systematic physical-world testing of autonomous driving systems[C] // 2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE). 2020: 347-358.
- [94] ARCAINI P, ZHANG X-Y, ISHIKAWA F. Targeting patterns of driving characteristics in testing autonomous driving systems[C] // 2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST). 2021 : 295–305.
- [95] HAUER F, PRETSCHNER A, HOLZMÜLLER B. Fitness functions for testing automated and autonomous driving systems[C] // International Conference on Computer Safety, Reliability, and Security. 2019: 69-84.

- [96] LI G, LI Y, JHA S, et al. AV-FUZZER: Finding safety violations in autonomous driving systems[C] // 2020 IEEE 31st International Symposium on Software Reliability Engineering (ISSRE). 2020: 25-36.
- [97] PEREIRA J L, ROSSETTI R J. An integrated architecture for autonomous vehicles simulation[C] // Proceedings of the 27th annual ACM symposium on applied computing. 2012 : 286–292.
- [98] MAMUN M A A, BERGER C, HANSSON J. MDE-based sensor management and verification for a self-driving miniature vehicle[C] // Proceedings of the 2013 ACM workshop on Domain-specific modeling. 2013 : 1-6.
- [99] FREMONT D J, KIM E, PANT Y V, et al. Formal scenario-based testing of autonomous vehicles: From simulation to the real world[C] // 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC). 2020: 1-8.
- [100] WITSCH A, OPFER S, GEIHS K. A formal multi-agent language for cooperative autonomous driving scenarios[C] // 2014 International Conference on Connected Vehicles and Expo (ICCVE). 2014 : 546-551.
- [101] DREOSSI T, FREMONT D J, GHOSH S, et al. Verifai: A toolkit for the formal design and analysis of artificial intelligence-based systems[C] // International Conference on Computer Aided Verification. 2019: 432-442.
- [102] CHANG M-F, LAMBERT J W, SANGKLOY P, et al. Argoverse: 3D Tracking and Forecasting with Rich Maps[C] // Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
- [103] INDAHENG F, KIM E, VISWANADHA K, et al. A Scenario-Based Platform for Testing Autonomous Vehicle Behavior Prediction Models in Simulation[J]. arXiv preprint arXiv:2110.14870, 2021.

# 简历与科研成果

基本情况 夏志龙,男,汉族,1997年2月6日出生于江苏南通

教育背景

2019年9月-2022年6月南京大学软件学院	硕士
2015 年 9 月-2019 年 6 月吉林大学软件学院	本科

读研期间的成果(包括发表的论文及参与的专利)

- 1. 冯洋, 夏志龙, 郭安, 等. 自动驾驶软件测试技术研究综述 [J]. 中国图象图形学 报, 2021.
- Ni Y, Xia Z, Zhao F, et al. An Online Multistep-Forward Voltage-Prediction Approach Based on an LSTM-TD Model and KF Algorithm[J]. Computer, 2021, 54(8): 56-65.
- 3. 基于可理解特征变异的智能交通数据测试样本生成方法 CN202010714316.2 已授权
- 4. 一种基于智能交通模型正确性的蜕变测试方法 CN202010712772.3 实质审查
- 5. 一种面向自动驾驶软件系统激光雷达的分布式模糊测试方法 CN202010714657.X 实质审查
- 6. 一种基于规则变异的自动驾驶激光雷达数据扩增方法 CN202010714660.1 实 质审查

### 读研期间参与的项目

- 国家自然科学基金项目(重点项目):智能软件系统的数据驱动测试方法与技术(61932012),2020-2024
- 国家自然科学基金项目(重点项目):面向安全攸关深度学习系统的软件测试 技术(61832009),2019-2023

致 谢

感谢我的导师陈振宇教授、责任导师冯洋助理研究员在论文研究和实验方 案设计时的帮助。陈振宇老师在3年中不仅给予我学习和科学研究上的指导,经 常主动为我答疑解惑,在生活上也十分关心我,在遇到困难时理解和照顾我。疫 情期间,两位老师在毕设指导工作上严谨负责,时常督促我认真实验,规范写 作,给与了我很大帮助。

感谢南京大学软件学院的各位其他老师3年以来的栽培,丰富的教学内容 和活动让我受益匪浅,让我掌握了更多的专业知识,在学校的日子里交到了很 多志同道合的朋友。

感谢我的室友朱晨乾、王越和步浩然同学, iSE 实验室的郭超、倪烨同学, 以及很多其他同学和朋友, 在大学生活和学习中时常帮助我解决各种问题, 在 毕业设计和论文撰写的日子里相互督促和鼓励。正因为有了他们, 我的大学生 活才充满了色彩。

最后感谢我的家人们,一直为我无私奉献,我将永远心怀感恩。