DriveFuzz:
Discovering Autonomous Driving Bugs through Driving Quality-Guided Fuzzing

Seulbae Kim\textsuperscript{1} \hspace{1cm} Major Liu\textsuperscript{2} \hspace{1cm} Junghwan Rhee\textsuperscript{3}
Yuseok Jeon\textsuperscript{4} \hspace{1cm} Yonghwi Kwon\textsuperscript{5} \hspace{1cm} Chung Hwan Kim\textsuperscript{2}

\textsuperscript{1}Georgia Institute of Technology, \textsuperscript{2}University of Texas at Dallas, \textsuperscript{3}University of Central Oklahoma, \textsuperscript{4}UNIST, \textsuperscript{5}University of Virginia
Can we trust autonomous driving systems?

• Expectation
  vs
  Reality

Actual bug we detected!
Can we trust autonomous driving systems?

• Fatal autopilot accidents continue
Finding bugs via manual testing

Source: “Will Tesla Autopilot hit a dog, human, or traffic cone?”
- Youtube Lowlifemike

Source: “Will a Tesla KILL a cat?”
- Youtube Carwow
Finding bugs via automated testing

- Feedback-driven fuzzing for traditional software

![Diagram showing the process of finding bugs via automated testing](image)

- Input
- Target system
- Code coverage
- Coverage feedback

Bug: segmentation fault
Finding bugs via automated testing

- Feedback-driven fuzzing for autonomous driving systems?
Layers and workflow of Autonomous Driving System (ADS)

Sensing
- Inputs: Environment, Vehicle States
- Sensor data

Perception
- Perceived states, nearby objects, ...

Planning
- Routing plan

Actuation
- Control commands

Map

Destination
Considerations in designing test inputs

The test input should not be a snapshot

The test input should be able to stress all layers
Our input space: Driving scenarios

- Representing temporal and spatial domains of real world
- Consists of
  - 3D map
  - Mission (initial and goal positions)
  - Actors (vehicles or pedestrians)
  - Puddles (e.g., black ice)
  - Weather conditions
Mutation of driving scenarios

- Map and mission selection
  - stress ADS with diverse environments
- Actor generation & mutation
  - render diverse interactive situations
- Puddle generation & mutation
  - stress planning & actuation layers with frictional diversity
- Weather mutation
  - affect sensing and perception
Confining mutation to feasible scenarios

- Ensuring physically valid mutation
  - Spatial constraint
    - Initial positions of all actors and objects are spread away (e.g., 5 m)
    - Preventing unrealistic jams (e.g., vehicles overlapping)
  - Temporal constraint
    - Maximum speed of actor vehicles and pedestrians are conservatively set
    - Preventing reckless behaviors (e.g., a person running into a vehicle too quickly)

- Both constraints are configurable
Feedback-driven fuzzing for ADS

Input scenario → ADS → ?

feedback
Defining bugs

• What happens to a buggy ADS?

Classic software bugs
Safety-critical vehicular misbehaviors

Collision

Infraction

Immobility
Feedback-driven fuzzing for ADS

Input scenario → ADS → ? feedback

Coverage feedback?
A need for a new feedback mechanism

General software programs

- Diverse, linear code paths
- More code paths $\approx$ more bugs found

Autonomous driving system

- Distributed system
- Behavior is driven by state changes in a loop, not code paths
A need for a new feedback mechanism

General software programs

- Diverse, linear code paths
- More code paths $\approx$ more bugs found

Autonomous driving system

- Distributed system
- Behavior is driven by state changes in a loop, not code paths

Need proper metrics to quantify the quality of input driving scenarios
Solution: Driving quality feedback

- Intuition
  - Quality of driving $\approx$ likelihood of misbehaviors

Hard acceleration, braking, and turns

- Metric auto insurance companies use

Oversteer and understeer

- #1 cause of motorsport accidents

Minimum distance to other actors

- Near-missed collisions
DriveFuzz overview

- **Input scenario**
  - Seed pool

- **Mutation engine (Section 4.2)**
  - Mission mutator
  - Weather mutator
  - Actor mutator
  - Puddle* mutator

- **Mutated scenario**

- **Test executor (Section 4.3)**
  - Test bridge
  - Driving simulator
  - Autonomous driving system (ADS)

- **Vehicle states (position, velocity, acceleration, ...)**

- **Misbehavior detector (Section 4.4)**
  - Driving test oracles
    - Collision
    - Infraction
    - Immobility

- **Driving quality feedback engine (Section 4.5)**
  - Quantifying driving quality via vehicle states
    - 1) Hard acceleration/braking
    - 2) Hard turn
    - 3) Over/understeer
    - 4) Minimum distance

*Puddle is invisible (It is visible in the illustrations for presentation)
DriveFuzz in action

- Seed scenario
  - Map
  - Initial position
  - Destination
## DriveFuzz in action

### Round 1

<table>
<thead>
<tr>
<th>Mutation #1</th>
<th>Mutation #2</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Mutation #1" /></td>
<td><img src="image" alt="Mutation #2" /></td>
</tr>
<tr>
<td>No misbehavior detected</td>
<td></td>
</tr>
<tr>
<td>score: 100</td>
<td>score: 100</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mutation #3</th>
<th>Mutation #4</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Mutation #3" /></td>
<td><img src="image" alt="Mutation #4" /></td>
</tr>
<tr>
<td>Check driving quality scores</td>
<td>SELECT</td>
</tr>
<tr>
<td>score: 100</td>
<td>score: 88</td>
</tr>
</tbody>
</table>
DriveFuzz in action

• Round 2

Mutation #4-1

Mutation #4-2

Misbehavior detected (collision)

Mutation #4-3

Save states and report
Evaluation

• Targeted two autonomous driving systems
  • Autoware
    • A full-fledged ADS with active development status
    • Internationally adopted by well-known auto manufactures (e.g., BMW)
    • Qualified to run driverless vehicles on public roads in Japan (2017~)
  • Behavior Agent
    • A rudimentary ADS developed by CARLA
    • Implements path planning and feedback-based PID control
    • Complies with traffic laws and avoids collisions
Detected 33 new bugs throughout ADS layers

<table>
<thead>
<tr>
<th>Bug #</th>
<th>Layer</th>
<th>Component</th>
<th>Description</th>
<th>Impact</th>
<th>Strategy</th>
<th>Root cause</th>
<th>ACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Sensing</td>
<td>Fusion</td>
<td>LiDAR &amp; camera fusion misses small objects on road</td>
<td>C</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>Perception</td>
<td>Detection</td>
<td>Perceives the road ahead as an obstacle at a steep downhill</td>
<td>I</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Perception</td>
<td>Detection</td>
<td>Fails to semantically tag detected traffic lights and cannot take corresponding actions</td>
<td>C, V</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>Perception</td>
<td>Detection</td>
<td>Fails to semantically tag detected stop signs and cannot take corresponding actions</td>
<td>C, V</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>Perception</td>
<td>Detection</td>
<td>Fails to semantically tag detected speed signs and cannot take corresponding actions</td>
<td>V</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>Perception</td>
<td>Localization</td>
<td>Faulty localization of the base frame while turning</td>
<td>C, L</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>Perception</td>
<td>Localization</td>
<td>Localization error when moving underneath bridges and intersections</td>
<td>C, L</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>Planning</td>
<td>Global planner</td>
<td>Generates infeasible path if the goal is unreachable</td>
<td>C, L</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>09</td>
<td>Planning</td>
<td>Global planner</td>
<td>Generates infeasible path if the goal’s orientation is not aligned with lane direction</td>
<td>C, I, L</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Planning</td>
<td>Global planner</td>
<td>Global path starts too far from the vehicle’s current location</td>
<td>C, I, L</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Planning</td>
<td>Local planner</td>
<td>Target speed keeps increasing at certain roads, overriding the speed configuration</td>
<td>S, C</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Planning</td>
<td>Local planner</td>
<td>Fails to avoid forward collision with a moving object</td>
<td>C</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Planning</td>
<td>Local planner</td>
<td>Fails to avoid lateral collision (ADS perceives the approaching actor before collision)</td>
<td>C</td>
<td>ent</td>
<td>Not impl</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Planning</td>
<td>Local planner</td>
<td>Fails to avoid rear-end collision (ADS perceives the approaching actor before collision)</td>
<td>C</td>
<td>ent</td>
<td>Not impl</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Planning</td>
<td>Local planner</td>
<td>While turning, ego-vehicle hits an immobile actor partially blocking the intersection</td>
<td>C</td>
<td>ent</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Actuation</td>
<td>Pure pursuit</td>
<td>Ego-vehicle keeps moving after reaching the destination</td>
<td>C, L</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Actuation</td>
<td>Pure pursuit</td>
<td>Fails to handle sharp right turns, driving over curbs</td>
<td>C, L</td>
<td>all</td>
<td>Faulty conf</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Perception</td>
<td>Detection</td>
<td>Indefinitely stops if an actor vehicle is stopped on a sidewalk</td>
<td>I</td>
<td>ent</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Perception</td>
<td>Detection</td>
<td>Flawed obstacle detection logic; lateral movement of an object is ignored</td>
<td>C</td>
<td>con</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Planning</td>
<td>Global planner</td>
<td>Generates inappropriate trajectory when initial position is given within an intersection</td>
<td>C, L, V</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Planning</td>
<td>Local planner</td>
<td>Improper lane changing, cutting off and hitting an actor vehicle</td>
<td>C</td>
<td>man</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Planning</td>
<td>Local planner</td>
<td>Vehicle indefinitely stops at stop signs as planner treats stop signs as red lights and waits for green</td>
<td>I</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Planning</td>
<td>Local planner</td>
<td>Vehicle does not preemptively slow down when the speed limit is reduced</td>
<td>S</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Planning</td>
<td>Local planner</td>
<td>Always stops too far (~ 10 m) from the goal due to improper checking of waypoint queue</td>
<td>F</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Planning</td>
<td>Local planner</td>
<td>Collision prevention does not work at intersections (only checks if actors are on the same lane)</td>
<td>C</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Planning</td>
<td>Local planner</td>
<td>Fails to avoid lateral collision (ADS perceives the approaching actor before collision)</td>
<td>C</td>
<td>man</td>
<td>Not impl</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Planning</td>
<td>Local planner</td>
<td>Fails to avoid rear-end collision (ADS perceives the approaching actor before collision)</td>
<td>C</td>
<td>man</td>
<td>Not impl</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Planning</td>
<td>Local planner</td>
<td>No dynamic replanning; the vehicle does infeasible maneuvers to go back to missed waypoints</td>
<td>C, L</td>
<td>ins</td>
<td>Not impl</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>Actuation</td>
<td>Controller</td>
<td>Keeps over-accelerating to achieve the target speed while slipping, creating jolt back on dry surface</td>
<td>C, L</td>
<td>ins</td>
<td>Not impl</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>Actuation</td>
<td>Controller</td>
<td>Motion controller parameters (PID) are poorly tuned, making the vehicle overshoot at turns</td>
<td>C, L</td>
<td>all</td>
<td>Faulty conf</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>Simulator</td>
<td></td>
<td>Simulation does not properly apply control commands</td>
<td>C, L, V</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>Simulator</td>
<td></td>
<td>Vector map contains a dead end blocked by objects as a valid lane</td>
<td>L, V</td>
<td>all</td>
<td>Data err</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>Simulator</td>
<td></td>
<td>Occasionally inconsistent simulation result</td>
<td>L, V</td>
<td>all</td>
<td>Logic err</td>
<td></td>
</tr>
</tbody>
</table>

[Impact] C: Collision / F: Fails to complete a mission / I: Vehicle becomes Immobile / L: Lane invasion / S: Speeding / V: Miscellaneous traffic Violation
The impact of driving quality feedback

- Fuzzing with and without driving quality feedback
  - Approximately 2x bugs detected with the feedback
An interesting bug

Multi-layer faults

• Sensing & Perception
  • Fails to perceive the puddle

• Planning
  • Fails to consider the slipping state
  • Keeps commanding speed-up

• Actuation
  • Missing Electronic Stability Control (ESC)
  • Keeps increasing the throttle amount
DriveFuzz summary

• DriveFuzz: End-to-end fuzzing framework for ADS
• Mutate driving scenarios
  • Mission, actors, puddles, weather
• Look for safety-critical misbehaviors
  • Collision, infraction, and immobility
• Leverage semantic feedback using driving quality metrics
• Found 30 bugs in two industry grade ADS
  • Readily exploitable by controlling nearby actors or objects
• Additional materials
  • Website & code: https://drivefuzz.autoinsight.dev/
Q & A