Trustworthy AI Autonomy
M4-1 Certification and Digital Twin Generation

Ding Zhao
Assistant Professor
Carnegie Mellon University

2022 @ Ding Zhao
Plan for today

- Importance of Certification/Evaluation
- Evaluation and test methods of AI autonomy
  - Naturalistic Field Operational Test (N-FOT) - Test on the public roads
  - Proving ground tests
  - Simulation/digital twins/augmented reality/meta universe
- Concept of Scenarios
THE FUTURE OF TRANSPORTATION STACK

263 Self-Driving Car Startups

SERVICES
- Route Planning
- Parking
- Car Hailing + Pooling
- Other: Aftermarket, Repair, Rental

IN-CAR INTELLIGENCE + ASSISTANCE
- Physical Car & Driver Safety + Accident Detection
- Emotion, Fatigue & Alcohol Detection + Distraction Avoidance
- Cybersecurity
- Intrusion, Tracking & Recovery

AUTONOMY
- Vehicle Diagnostics & Predictive Maintenance + Sensor-Based Vehicle Safety
- Passenger-Focused Sensors (Including Usage-Based Insurance)
- Infotainment + Display
- Personal / Voice Assistance
- Navigation Assistance + Pedestrian Analysis & Communications

AUTONOMOUS VEHICLE MAKER + TOOLS
- Autonomous Vehicle Maker + Tools

INFRASTRUCTURE + CONNECTED CAR
- Sensor Networking Infrastructure (V2V, V2I, 5G, Cellular, Wi-Fi)
- Connected Car - Data, Platform, Software
- Fleet + Traffic Management
- OTA CAR SOFTWARE UPGRADE + SMART PHONE INTEGRATION

FLIGHT
- FLIGHT

INTELLIGENT MANUFACTURING
- New/Advanced Materials
- Rapid Prototyping - 3D Printing, Modularization, Open Source
- Advanced / Automated Assembly Line
- Material Characterization & Testing

ONBOARD SENSORS
- Location - GIS, Precision Positioning, Path Planning
- Vision / Camera
- LiDAR
- Radar
Regulations

Automated Vehicles 3.0

PREPARING FOR THE FUTURE OF TRANSPORTATION

October 2018

With the development of automated vehicles, American creativity and innovation hold the potential to once again transform mobility.

Planned and Operational Connected Vehicle Deployments

Where Infrastructure and In-Vehicle Units are Planned or In Use

- Planned Projects
- Operational Projects

Source: USDOT September 2018

- Operational (52 Projects)*
  - Planned (23 projects)*: **

<table>
<thead>
<tr>
<th></th>
<th>Infrastructure Units</th>
<th>In-Vehicle Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational (52 Projects)*</td>
<td>2,044</td>
<td>3,340</td>
</tr>
<tr>
<td>Planned (23 projects)*: **</td>
<td>242</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>2,286</td>
<td>3,340</td>
</tr>
</tbody>
</table>

* Projects shown include those sponsored by U.S. DOT and others.
** Device numbers for many of the planned projects are currently unavailable.
New regulation

DEPARTMENT OF TRANSPORTATION
National Highway Traffic Safety Administration
49 CFR Part 571
Docket No. NHTSA-2021-0003
RIN 2127-AM06
Occuant Protection for Vehicles With
Automated Driving Systems

AGENCY: National Highway Traffic Safety Administration (NHTSA), Department of Transportation.

ACTION: Final rule.

SUMMARY: This final rule amends the occupant protection Federal motor vehicle safety standards (FMVSSs) to account for future vehicles that do not have the traditional manual controls associated with a human driver because they are equipped with Automated Driving Systems (ADS). This final rule makes clear that, despite their innovative designs, vehicles with ADS technology must continue to provide the same high levels of occupant protection that

U.S. clears way for truly driverless vehicles without steering wheels

PUBLISHED FRI, MAR 11 2022•10:31 AM EST | UPDATED FRI, MAR 11 2022•1:04 PM EST

Michael Wayland
@MIKewayLAND

KEY POINTS
* Federal vehicle safety regulators have cleared the way for the production and deployment of truly driverless vehicles that do not include manual controls such steering wheels or pedals.

* The U.S. National Highway Traffic Safety Administration on Thursday issued final rules eliminating the need for highly automated and self-driving vehicles to need such controls.

* The new rule emphasizes such cars "must continue to provide the same high levels of occupant protection as current passenger vehicles."

Certification and Evaluation methods

Efficient Testing

- On-sim dynamic test
- Worst-case test
- Predefined test
- On-sim predefined test

Rich Information

On-track/Proving ground
- On-track dynamic test
- On-track + AR dynamic test

On-road/Field operational test
- On-road autonomous driving test
- On-road human-driven driving test

AV sequential deployment plan

EURO NCAP scenario list

On-sim/Simulation

- On-sim dynamic test

On-track/Proving ground

Resource Consumption

Fidelity Level
How safe is safe enough for AVs?

How to measure the safety: Simulation, Vehicle in-the-loop simulation (VIL), physical tests, Open-roads
How safe is safe enough for AVs?

Table 13: Distribution of scenarios by testing approach.
Naturalistic Field Operational Tests (NFOT)

Waymo’s self-driving car performing left-turn maneuver
AV Deployment

As of June 20, 2018, there are 56 Autonomous Vehicle Testing Permit holders.

Source: DMV.ca.gov (https://www.dmv.ca.gov/portal/dmv/detail/vr/autonomous/testing)

- AV testing in California

Source: DMV.org (https://www.dmv.org/articles/top-5-cities-for-self-driving-boom)

San Jose, CA  Ann Arbor, MI  Boston, MA  Pittsburgh, PA  Austin, TX
Naturalistic Field Operational Tests (NFOT)

Waymo Disengagement Rate (Per 1000 Miles)

Waymo, Kirkland, Mountain View, Daly City, Half Moon Bay, Los Altos, Menlo Park, Merced, Morgan Hill, San Francisco, San Luis Obispo, Santa Cruz, Sunnyvale, Palo Alto, Tiburon, Truckee, CA, AZ, TX, MI, WA, GA, Atlanta, Detroit, Phoenix, Chandler, Gilbert, Guadalupe, Mesa, Tempe

20 million miles and counting...

Forbes, January 2020
Driving Datasets for Autonomous Vehicles
Comparison to human baselines

Classification of Waymo-involved collisions (6.1 million AV miles driven)

A total of 18 dangerous situations observed during data collection and 29 situations during simulation

---

Proving grounds tests

• Proving grounds provide a physical semi-controllable environment to test AI, e.g. autonomous vehicles (AV)

• Example of AV proving ground facilities:
  • Mcity (UM)
  • The Castle (Waymo)
  • ALMONO (Uber)
  • American Center for Mobility
  • SMART Transportation Research Center (US DOT)
  • Kcity (South Korea)
Proving grounds

- Geometric based test scenario generations
Proving grounds

• Data-driven test scenario generations

• Key steps:
  
  • Extract driving scenarios from driving database
  
  • Extract proving ground map geometries and assets
  
  • Optimize the scenario-map compatibility

Simulations

- Simulations allow fast and fully-controlled testing for AI algorithms
- Simulation-based testing is often done at various fidelity level

Simulation for Urban Mobility (SUMO)  
Scalable Multi-Agent Reinforcement Learning Training School (SMARTS)  
Highway Gym Environment (highway-env)  
CARLA simulation  
WeBot for Automobiles  
PreScan  
Uber ATG simulation platform
Waymo simulated collisions

**Head-on collisions**

- Other vehicle travelling at 38 mph
- Simulated vehicle travelling at 29.5 mph
- Simulated stopping location
- Simulated full braking initiated
- Waymo travelling in AV mode (36 mph)

**Rear-end collisions**

- Waymo braking to stop at traffic light (5 mph)
- Other vehicle travelling at 25 mph
- Waymo braking to stop at traffic light (27 mph)
- Other vehicle travelling at 26 mph

Synthesis tests

- Multi-fidelity models (e.g. Gaussian processes) are promising to synthesize information among various testing modes

- Historical data

- AR test

- On-road test

- Test resources under varying fidelity $t = 1, 2, ..., T$

- Various test scenario $x \in \mathcal{X}$

- Performance data from test model $h_t(x)$

- Kriging-based response surface model, $y_t(x)$

- Target model $g(x)$

- Multi-fidelity model, $y_T(x)$

Huang "Synthesis of Different Autonomous Vehicles (AV) Test Approaches", ITSC, 2018
Evaluation and test methods for AI autonomy

Efficient Testing

Fidelity Level

On-sim/Simulation

On-track/Proving ground

Rich Information

Multi-fidelity model

On-road/Field operational test

On-road autonomous driving test

On-road human-driven driving test

On-track dynamic test

On-track predefined test

On-sim dynamic test

On-sim predefined test

Resource Consumption

EURO NCAP scenario list

AV sequential deployment plan

Worst-case test

Predefined test

Efficient Testing

Huang “Synthesis of Different Autonomous Vehicles (AV) Test Approaches”, ITSC, 2018
Methods

- Traditional ways to identify scenarios
- Data-based Scenario Generation
- Adversarial Scenario Generation
- Knowledge-based Scenario Generation
Traditional ways to identify scenarios

Table 5. Priority V2V Pre-Crash Scenarios

<table>
<thead>
<tr>
<th>No</th>
<th>Pre-Crash Scenario</th>
<th>Group</th>
<th>Cost</th>
<th>FYL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SCP @ non signal</td>
<td>Junction Crossing</td>
<td>20.4%</td>
<td>20.7%</td>
</tr>
<tr>
<td>2</td>
<td>LTAP/OD</td>
<td>LTAP/OD</td>
<td>15.1%</td>
<td>15.3%</td>
</tr>
<tr>
<td>3</td>
<td>Rear-end/LVS</td>
<td>Rear-End</td>
<td>14.8%</td>
<td>14.0%</td>
</tr>
<tr>
<td>4</td>
<td>Opposite direction/no maneuver</td>
<td>Opposite Direction</td>
<td>14.7%</td>
<td>15.1%</td>
</tr>
<tr>
<td>5</td>
<td>Rear-end/LVD</td>
<td>Rear-End</td>
<td>6.1%</td>
<td>5.8%</td>
</tr>
<tr>
<td>6</td>
<td>Rear-end/LVM</td>
<td>Rear-End</td>
<td>5.1%</td>
<td>5.1%</td>
</tr>
<tr>
<td>7</td>
<td>Changing lanes/same direction</td>
<td>Lane Change</td>
<td>4.2%</td>
<td>4.3%</td>
</tr>
<tr>
<td>8</td>
<td>Turning/same direction</td>
<td>Lane Change</td>
<td>3.1%</td>
<td>3.0%</td>
</tr>
<tr>
<td>9</td>
<td>Opposite direction/maneuver</td>
<td>Opposite Direction</td>
<td>1.7%</td>
<td>1.8%</td>
</tr>
<tr>
<td>10</td>
<td>Drifting/same direction</td>
<td>Lane Change</td>
<td>1.7%</td>
<td>1.8%</td>
</tr>
</tbody>
</table>

Total: 86.9% 87.0%

Limitation

- Scenarios manually selected by human may not be able to take the advantage of the big data.
- Human and AVs may have different critical scenarios.

**EURO NCAP AEB**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$v_L(t_0)$ [km/h]</th>
<th>$a_L$ [m/s$^2$]</th>
<th>$R_L$ [m]</th>
<th>$v(t_0)$ [km/h]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>30:5:80</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>0</td>
<td>100</td>
<td>30:5:70</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>-2 &amp; -6</td>
<td>12 &amp; 40</td>
<td>50</td>
</tr>
</tbody>
</table>

---

**Static**  
**Moving**  
**Braking**
Realistic safety-critical scenario generation

• Opportunities:
  • Cheaper data access
  • More powerful computational facilities
  • Better machine learning algorithms

• Challenges:
  • Data sparsity /imbalance/rarity
  • Multi-modes
  • Dynamic long-horizon temporal decision making
  • High dimensional sensing input
Why safety-critical scenarios?

A new stage of evaluating autonomous vehicles: safety-critical scenarios

* Data source: California Department of Motor Vehicle disengagement report 2020
How to get safety-critical scenarios?

Collect from real-world road test
- Tremendously rare
- Expensive

Collect from rule-based simulators
- Lack of diversity
- Not realistic

What’s the expectation of a good scenario generator?
- Reality, Adaptability, Controllability, Efficiency, Diversity
Data-based Scenario Generation

- What does each latent variable mean? How to get the trajectories we want?
- Safety-critical data is still rare in the latent space.
Data-based Scenario Generation

Training Process

Real-world Trajectories

Encoder

Decoder

Reconstructed Trajectories

Generating Process

Randomly sample from the learned latent space

Sampled Trajectories

W. Ding, W. Wang, D. Zhao, A new multi-vehicle trajectory generator to simulate vehicle-to-vehicle encounters, ICRA 2019
Data-based Scenario Generation

Assume encounter trajectories are formed by two parts: road shape (style), risk level (content)
Driving scenario generation
Generative models

A generative model is a neural network that takes a unit Gaussian distribution as input and outputs a distribution of data samples. The loss is calculated by comparing the generated distribution with the true data distribution.

Deconvolution operations:
- Transpose convolution: expanding the input with intermediate grid

Output size = (Input size - 1) / stride + 1 + kernel size

Ding Zhao | CMU | 2021

https://openai.com/blog/generative-models/
Vanilla autoencoder

- $x = d(e(x))$: Lossless encoding, no information is lost when reducing the number of dimensions.
- $x \neq d(e(x))$: Lossy encoding, some information is lost when reducing the number of dimensions and can't be recovered later.
Linear autoencoder

- The optimal solution of a linear autoencoder can be obtained with PCA (Principal Component Analysis). The latent space will be obtained by calculating the Singular Value Decomposition (SVD).

\[
X \rightarrow e(x) = P^T x \quad d(e(x)) = PP^T x
\]

<table>
<thead>
<tr>
<th>Point</th>
<th>Initial</th>
<th>Encoded</th>
<th>Decoded</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>(-0.50, -0.40)</td>
<td>-0.63</td>
<td>(-0.54, -0.33)</td>
</tr>
<tr>
<td>B</td>
<td>(-0.40, -0.10)</td>
<td>-0.39</td>
<td>(-0.34, -0.20)</td>
</tr>
<tr>
<td>C</td>
<td>(0.10, 0.00)</td>
<td>0.09</td>
<td>(0.07, 0.04)</td>
</tr>
<tr>
<td>D</td>
<td>(0.30, 0.30)</td>
<td>0.41</td>
<td>(0.35, 0.21)</td>
</tr>
<tr>
<td>E</td>
<td>(0.50, 0.20)</td>
<td>0.53</td>
<td>(0.46, 0.27)</td>
</tr>
</tbody>
</table>
Autoencoder

- Autoencoder can be used as data compression algorithm
- Google+ sends “latent images” and uses auto encoder to reconstruct images locally

[Image: Original image of a surfer, compared to a RAISR-enhanced version. Original size 1000x1500, 100kb. RAISR size is 1000x1500, 25kb.]

Instead of requesting a full-sized image, G+ requests just 1/4th the pixels...

...and uses RAISR to restore detail on device

[Link: https://www.slrlounge.com/google-raisr-image-resolution-enhancement-straight-out-of-csi/]
Autoencoder with neural networks
Issues of unregulated autoencoder

- Question: can we use one dimensional number in the latent space?
- Two ideas:
  - 1) add noise to randomize the system; 2) regularize the latent space

(encoded data can be decoded without loss if the autoencoder has enough degrees of freedom)
(point sampled from the one dimensional latent space for new content generation)
(without explicit regularisation, some points of the latent space are “meaningless” once decoded)
Variational Autoencoder

\[ \text{loss} = \| x - \hat{x} \|^2 + \text{KL}[ N(\mu_x, \sigma_x), N(0, I)] = \| x - d(z) \|^2 + \text{KL}[ N(\mu_x, \sigma_x), N(0, I)] \]

<table>
<thead>
<tr>
<th>Simple Autoencoders</th>
<th>Variational Autoencoders</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input: ( x )</td>
<td>Input: ( x )</td>
</tr>
<tr>
<td>Encoding: ( z = e(x) )</td>
<td>Encoding: ( p(z</td>
</tr>
<tr>
<td>Latent Representation: ( z )</td>
<td>Sampling: ( z \sim p(z</td>
</tr>
<tr>
<td>Decoding: ( d(z) )</td>
<td>Decoding: ( d(z) )</td>
</tr>
</tbody>
</table>

[Diagram of Variational Autoencoder](https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73)
Variational Autoencoder

\[ x \]

\[ \mu_x = g(x) \]
\[ \sigma_x = h(x) \]
\[ \zeta \sim N(0, I) \]

\[ z = \sigma_x \zeta + \mu_x \]
\[ \hat{x} = f(z) \]

loss = \[ C \| x - \hat{x} \|^2 + KL[ N(\mu_x, \sigma_x), N(0, I) ] = C \| x - f(z) \|^2 + KL[ N(g(x), h(x)), N(0, I) ] \]
Disentangled VAE ($\beta$-VAE)

- Goal: ensuring each dimension of latent vectors learn distinct attributes

- This can be achieved by adding hyperparameter $\beta$ to the loss function:
  \[ \mathcal{L}(\theta, \phi, \beta, X, Z) = \mathbb{E}[\log p_\theta(X | Z)] + \beta D_{KL}(q_\phi(Z | X) || p(Z)) \]

- The model then learns to use latent space as efficient as possible

Examples of VAE in practice

\[ \beta \text{-VAE} \quad \text{VAE} \]

(a) Azimuth (rotation)

(b) emotion (smile)

Visualize the latent code

- **PCA**: Principal Component Analysis
  - Linear method, not robust to outliers
- **t-SNE**: t-distributed stochastic neighbor embedding
  - Nonlinear method, slow, may apply PCA first
- **LDA**: Linear Discriminant Analysis
  - Maximize the separation between multiple classes
  - Fast, need to know the labels of classes
LDA: Linear Discriminant Analysis

**PCA:**
component axes that maximize the variance

**LDA:**
maximizing the component axes for class-separation

t-distributed stochastic neighbor embedding (t-SNE)

How to visualize the latent space $z$ of VAE?

• Reduce the dimension to 2 or 3
• Unsupervised dimension reduction
• Similar vectors should be close

Toolbox
• sklearn.manifold.TSNE

https://distill.pub/2016/misread-tsne/
t-distributed stochastic neighbor embedding (t-SNE)

How to visualize the latent space $z$ of VAE?
- Reduce the dimension to 2 or 3
- Unsupervised dimension reduction
- Similar vectors should be close

Toolbox
- `sklearn.manifold.TSNE`

https://www.oreilly.com/people/cyrille-rossant/
Safety-critical trajectory via VAE

- Use linear interpolation of collision data and safe data to generate safety-critical data
- Use road bird-view image as constraints
Safety-critical trajectory via VAE

- Different map conditions have different trajectory output
- $\lambda$ controls the risk value
Flow-based generative models

Approximate likelihood

- **VAE:** maximize ELBO.

Exact likelihood

- **Flow-based generative models:** minimize the negative log-likelihood

![Diagram showing the flow-based generative models process](https://lilianweng.github.io/lil-log/2018/10/13/flow-based-deep-generative-models.html)
Safety-critical scenarios generation with flow-based model

- An adversarial attack framework
- Use flow-based model to estimate the distribution of risky traffic scenarios
Safety-critical scenarios generation with flow-based model

Data-based Scenario Generation

Summary

- Use the real-world data
- Some kind of controllability

- Poor adaptivity, no interaction with downstream task
- Only use existing data, lack of diversity
Worth Reading

• Beta-VAE  
  Higgins, Irina, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, 
  Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. "beta-vae: 
  Learning basic visual concepts with a constrained variational 
  Check open review: https://openreview.net/forum?id=Sy2fzU9gl

• General intro to GAN:  
  Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, 
  Communications of the ACM, 63(11), pp.139-144. 
  GAN Lab: https://poloclub.github.io/ganlab/