5. Results and discussion

5.1 Comparative metrics study

The proposed method requires inputs from the simulator developers and the users to validate the applicability of the score. The Base score of simulators of the proposed parameters can be populated from the information provided from the documentation of the simulator. However, some parameters like Photorealism are subjective and cannot be fetched from the distinct sources. For evaluation these parameters these parameters are scored based on average results from various people. For user weights, two users who use simulators for different use case are identified and the parameters are scored based on the subjects’ perspective. The 3 simulators considered are CARLA [23], Summit [24], LGVSL [25]. The base score of these simulators can be found in the appendix. The Base score is populated from the information provided by the official Documentation.

The use case of the subjects used for evaluation is as follows

User1: User is a research student. Interested mainly on sensory data generated, feedback and training a DL model. Degree of Realism is not so important but would be an advantage

User2: User is a Software tester at an automobile company working on validation of model performance in real world.

The user weight of the parameters of these users can be found in the appendix. Table 3 summarizes the results from the evaluation. The score in the table corresponds to the final score which can be directly used for comparison.

|  |  |  |  |
| --- | --- | --- | --- |
|  | CARLA | Summit | LGVSL |
| User 1 | 18.5 | 21 | 19.7 |
| User 2 | 22.5 | 34 | 36 |

Table 3: Final scores of comparisons

The score shows that from the 3 simulators considered LGVSL suits well to the user 2 and SUMMIT appeals the most to User 1. This co relates with the fact that LGVSL provides various facility to integrate real world components which should be crucial for Use case of user 2 and Summit can deliver realistic traffic behaviours and as it is built on top of Carla, this can take the advantage of platforms provided by Carla to train ML algorithms. However, there should be a Mandatory parameter criterion where a parameter is mandatory for a user and if any of the simulators considered for evaluation doesn’t offer it, it should be excluded from the evaluation. This can aid the user to define some MUST criteria.

5.2 Generative model-based simulators

The proposed architecture was trained for 50 epochs and the on the mini batch size of 16. Figure 20 show the loss of generator and discriminator over the training.

Figure 20: Generator and Discriminator loss

The Mean square loss weight parameter alpha is reduced to half for every 10 epochs. By doing this, the generator is initially trained to reconstruct the data and gradually learning to generate new features.

The model has tested with various tailored made action commands and checked for its performance. Figure 21 shows the generation of next time step Observation Og t+1 for randomly selected action vector at on a prior time step Ot.

Figure 21: Transition of observation for the action command {Turn angle, Turn direction, velocity}

The shift peaks with in the signal proves that the observation transit of next time step in accordance with the action command. Figure 22 displays the generated output and the expected output for same action command

Figure 22: Generated output and the expected output

The generated output had some noise and it is not as clean as inputs sample from training corpus. However, no traces of new object generation are observed in the input. Discussion

The failure of generation of new objects is because the lack of diverse objects within the training data. Moreover, it is difficult to identify the generation of new plausible in 2d lidar data by visual inspection.

When these generations occur recursively i.e. the generated output Og t+1 is fed as input to the next time step. Over the time the noise within the generated data which represent the environment accumulates resulting in worst representation of environment. Figure 22 shows the observations generated recursively for 10-time steps.

Figure 22: Recursive generation of Sensor observations

The quality of the outputs are good for few time steps and after 5 steps the quality distorted.

CLEAN TEXT

5. Results and Discussion

5.1 Comparative Metrics Study

The methodology proposed for assessing simulator suitability necessitates inputs from both simulator developers and users to validate the applicability of the derived scores. While the base score of simulators for proposed parameters can largely be populated from official documentation, subjective parameters like "Photorealism" require alternative evaluation methods. To assess such subjective parameters, a collective perspective was gathered from various individuals. This included an average evaluation from multiple sources to ensure a well-rounded perspective.

To assess user weights, two distinct user personas were identified, each with unique use cases for simulator:

User 1: A research student primarily focused on sensory data, feedback, and training deep learning models. While realism holds importance, it's not a primary criterion.

User 2: A software tester at an automobile company dedicated to real-world model performance validation.

Three simulators CARLA [23], Summit [24], and LGVSL [25] were considered for evaluation, with their base scores sourced from official documentation (Appendix).

The derived user weights for the identified personas (found in the appendix) were utilized to calculate final scores, summarized in Table 3:

|  |  |  |  |
| --- | --- | --- | --- |
|  | CARLA | Summit | LGVSL |
| User 1 | 18.5 | 21 | 19.7 |
| User 2 | 22.5 | 34 | 36 |

Table 3: Final scores of comparisons

Interpreting the results, LGVSL emerges as a suitable choice for User 2, emphasizing the integration of real-world components, an essential criterion for their use case. Conversely, Summit resonates with User 1 due to its adeptness in simulating realistic traffic behaviours and leveraging CARLA's platform for training ML algorithms.

A critical suggestion for refinement involves the introduction of a mandatory parameter criterion. This criterion would empower users to define indispensable parameters, and if a simulator under evaluation fails to offer them, it should be excluded from the assessment. This approach ensures that users can establish essential criteria tailored to their specific needs.

This comparative analysis showcases the varied suitability of simulators based on distinct user perspectives, emphasizing the importance of tailored assessments for specific use cases.

5.2 Generative Model-Based Simulators

The proposed architecture underwent training for 50 epochs utilizing a mini-batch size of 16. Figure 20 illustrates the dynamic trends in generator and discriminator loss during the training phase.

Figure 20: Generator and Discriminator Loss

To optimize learning, the mean square loss weight parameter alpha was systematically reduced by half every 10 epochs. This strategy allowed the generator to initially focus on data reconstruction, progressively advancing to generate novel features.

Performance evaluation of the model involved rigorous testing with various customized action commands. Figure 21 portrays the generation of subsequent observation (Og t+1) based on randomly chosen action vectors at a preceding time step (Ot).

Figure 21: Transition of Observation for the Action Command {Turn angle, Turn direction, Velocity}

Evident shifts within the signal patterns validate the successful transition of observations in concordance with the designated action commands. Figure 22 provides a comparative analysis between the generated output and the expected output for identical action commands.

Figure 22: Generated Output and Expected Output

While the generated output exhibited discernible noise and lacked the pristine quality of input samples from the training corpus, no instances of new object generation were identified within the input.

The failure to generate new objects can be attributed to the limited diversity of objects within the training dataset. Furthermore, visually discerning the generation of plausible new objects within 2D lidar data posed significant challenges.

Recursive generation of outputs, wherein the generated output (Og t+1) serves as input for subsequent time steps, led to a cumulative increase in noise. Subsequently, this cumulative noise adversely impacted the fidelity of the environment's representation over time, as depicted in Figure 23, displaying observations generated recursively for 10 time steps.

Figure 23: Recursive Generation of Sensor Observations

Initial outputs exhibited commendable quality; however, the quality notably degraded after 5 steps.