**Report on Autonomous Driving in Carla Using DQN**

**Introduction to DQN**

Deep Q Networks (DQN) represent a neural network architecture employed in reinforcement learning (RL) that integrates Q-learning with deep learning techniques. Unlike traditional Q-learning, DQNs approximate Q-values, making them well-suited for complex and high-dimensional environments. The utilization of experience replay in DQNs prevents overfitting and stabilizes training, rendering them powerful for learning policies in model-free approaches.

**Network Inputs**

Driving DQN

d (Distance to Lane's Center): Represents the distance of the ego vehicle to the center of the lane.

ϕ (Angular Deviation): Indicates the angular deviation of the ego vehicle.

Braking DQN

d(obs) (Distance to Nearest Obstacle): Denotes the distance to the nearest obstacle.

v (Current Velocity): Represents the current velocity of the ego vehicle.

**Training Summary**

Map Routes: 3

Episodes: 40

Optimizer Learning Rate (LR): 3e-4

Discount Factor (Γ): 0.99

Epsilon (ε) at Start: 0.5

**Merits**

* Simple Architecture: DQN employs a straightforward architecture, facilitating ease of understanding and implementation.
* Fast to Train: The training process for DQNs is relatively fast, enabling quicker iterations and model refinement.
* Less Dependent on Hardware Resources: DQNs can be trained with less dependence on high-end hardware resources.
* Generalizes Well for Unseen Routes: The DQN model demonstrates a capability to generalize well for routes not encountered during training.

**Drawbacks**

* Discrete Action Space: DQN relies on a discrete action space, limiting the granularity of actions the model can take.
* Uses Only Waypoints for Navigation: The model's reliance on waypoints for navigation may restrict its adaptability to more dynamic environments.
* Does Not Utilize Images from Cameras for Training: The model does not take advantage of visual data from cameras, potentially limiting its ability to respond to visual cues in the environment.

**Conclusion**

The DQN-based approach presented in this report employs only two parameters, distance to the lane's center (d) and angular deviation (ϕ), for training the driving network. These parameters are relative to the nearest waypoint projected on the lane's center. While effective within Carla's simulator environment, this approach exhibits low scalability for real-world deployment due to its limited reliance on visual data and discrete action spaces. Future endeavors may involve exploring hybrid approaches that incorporate visual information for improved real-world adaptability.