Progress Report: Traffic RL Intersection RL Environment Visualization

Date: October 15, 2025

Things Completed-

- **Core Simulation Environment Integration**
  Have successfully integrated the custom 'TrafficIntersectionEnv' into a full visual loop using Pygame. The environment reflects the current phase of the traffic signal (GREEN_NS, YELLOW_NS, GREEN_EW, YELLOW_EW) and updates based on actions taken by the agent. Each 'env.step(action)' call advances the environment, controlling both pedestrian and vehicle flows. A random agent is currently used to simulate decisions, with reward accumulation enabled for future policy benchmarking.

- **Signal Phase & Timer Display**
  Signal phase and remaining timer are rendered cleanly in the top-left corner. Each phase dynamically updates, visually reflecting real-time state transitions. Signals are directional: NORTH, EAST, SOUTH, and WEST, corresponding to how cars and pedestrians flow through the intersection.

- **Vehicle Visualization with Entry and Movement Animation**
  Each direction spawns cars that queue before the intersection. Their position and movement reflect: Waiting cars in queues with spacing and active crossing cars shown with animated forward movement and arrow indicators.

- **Pedestrian Movement & Queueing**
  Pedestrians are spawned per direction with appropriate sidewalk positioning. Once a walk signal is available, they animate across the crosswalk with realistic spacing. Both entry and crossing animations have been implemented, with position calculated using dynamic interpolation (progress percentage-based).

- **Visual Enhancements & UI Design**
  Crosswalks render with white dashed lines for visual realism. Cars are shown in red with rounded rectangles. Pedestrians are yellow circular dots for visibility. A consistent road + intersection layout has been built to serve as a stage for signal control learning.