Driving Simulator

Level Documentation

**Base Level Class**

The level class stores a data representation of all the stationary parts of a level: roads, stop signs, decoration, etc. It also handles rendering of the level to the screen. This documentation explains all methods that are meant to be interfaced with. There are additional internal methods that are not documented.

**constructor:**

Level(width, height) -> returns an empty level with the specified dimensions. Level assets are added with the methods specified below.

**Class Constants:**

Level.BG\_COLOR: 3-tuple specifying the RGB color of the background.

Level.LANE\_COLOR: 3-tuple specifying the RGB color of the roads.

Level.Y\_LINE\_COLOR: 3-tuple specifying the RGB color of yellow road lines.

Level.W\_LINE\_COLOR: 3-tuple specifying the RGB color of white road lines.

Level.LANE\_WIDTH: width of each road lane, in pixels

Level.LINE\_WIDTH: width of road lines, in pixels

Level.STUB\_ROAD\_LEN: length of "stub roads” that come out of intersections,in pixels

Level.STOP\_LINE\_WIDTH: width of solid white lines marking where to stop at an intersection, in pixels.

Level.DASH\_LEN: dash size and spacing for dashed road lines, in pixels.

**LevelInstance.draw(target, x, y, debug=false)** -> draws the level on the specified target surface. X and Y represent the offset to draw the level at: the pixel at (x, y) in the level is drawn in the top left.

**LevelInstance.add\_horizontal\_road(start\_x, end\_x, y)** ->

Adds a horizontal 2-lane road to the level. The road spans from start\_x to end\_x and is centered at a height of y.

**LevelInstance.add\_vertical\_road(x, start\_y, end\_y)** ->

Adds a vertical 2-lane road to the level. The road spans from start\_y to end\_y and is centered at horizontal position x.

**LevelInstance.add\_diagonal\_road(start\_x, start\_y, end\_x, end\_y)** ->

Adds a 2-lane road to the level. The road spans from (start\_x, start\_y) to (end\_x, end\_y).

**LevelInstance.add\_4\_way\_stop(x, y)** ->

Adds a 4-way stop (2-lanes each way) centered at (x, y). The total width of the intersection is 2\*LANE\_WIDTH + 3\*LINE\_WIDTH + 2\*STUB\_ROAD\_LEN px.

**LevelInstance.add\_4\_way\_light(x, y)** ->

Adds a 4-way stoplight (2-lanes each way) centered at (x, y). The total width of the intersection is 2\*LANE\_WIDTH + 3\*LINE\_WIDTH + 2\*STUB\_ROAD\_LEN px.

**LevelInstance.add\_4\_lane\_divided(x1, y1, x2, y2)** ->

Adds a 4-lane divided highway spanning from (x1, y1) to (x2, y2) to the level.

**LevelInstance.add\_4\_lane\_with\_on\_ramp(x1, y1, x2, y2, ramp\_x, ramp\_y, join\_loc = 0.5)**

->

Adds a 4-lane divided highway spanning from (x1, y1) to (x2, y2) to the level. Additionally adds a 1-lane on-ramp, starting at (ramp\_x, ramp\_y) and ending at a fractional distance along the highway. That fractional distance is given by join\_loc.

**LevelInstance.add\_random\_decorations(n)** ->

Attempts to add n random decorations (grass or flowers) to the level. Decorations that are blocked by existing road elements will not be added.

**LevelInstance.add\_t\_intersection(x, y, entrance= ”bottom”)** ->

Adds a T-intersection to the level, centered at (x, y). Entrance is a string describing the side at which the secondary road enters the main road of the T. It can be one of “bottom”, “top”, “left”, or “right”. The main road is vertical if left or right, and horizontal if top or bottom.

**LevelInstance.get\_targets(sprite)** ->

Returns a list containing any RoadLanes that the given sprite is colliding with. Used for lane detection. The RoadLane class shouldn’t need to be interfaced with directly. The only method that you should need is RoadLaneInstance.get\_center(), which returns the coordinates of the center of the lane and can be used as a unique identifier.

**Tutorial**

**Level 1**

**Level 2**

**Level 3**

In this level, the player is coming up on a four way stop. There are 3 AI cars that pull up to the stop signs. One coming the horizontal road opposite from the player car. Two cars are going two different ways on the vertical road. The AI cars stop and go at the stop sign. The players objective is to make the right road decision so he or she will pass the level.

The player will fail if they crash into one of the other cars, drive on the wrong side of the road, or drive through the stop sign without stopping.

The player will pass the level if he or she successfully stops at the stop sign and goes straight or left or right on the road.

**Level 4**

In this level, the player is on a horizontal two-way road traveling right. An AI car is traveling in the other lane toward the player. As they approach, the AI car turns into the player’s lane. The player is expected to turn and/or break, while not turning into the other lane, in order to avoid colliding with the AI car.

The player will fail if they:

* Do not get up to speed within about 3 seconds
* Touch the other lane
* Go totally offroad before the AI car has turned toward them
* Collide with the AI car

The player passes if they do not trigger any fail conditions, passing about 3 seconds after the AI car turns towards the player.

**Level 5**

In this level, the player begins on the on-ramp of a 4-lane divided highway. A constant stream of AI car traffic is spawned on both lanes of the highway, and the player must successfully accelerate down the on-ramp and merge onto the highway.

The player will fail the level if they:

* Go offroad or between lanes for more than one second
* Collide with another car
* Go past the onto or past the highway’s median

The player passes if they do not trigger any fail conditions and successfully drive off the right side of the screen after merging onto the highway.