3-D Simulator Programming Competition

Competition Synopsis

The motivation for this competition is simple. Testing search algorithms and other “mission” logic currently requires a fully functioning ideal vehicle. Since this is not always available, a method for testing “artificial intelligence” code is required for faster/parallel development. This type of simulator is also paramount for the success of real world missions that must be deployed quickly into unknown environments. The mission logic and search algorithms must prove to be very robust if there is any hope of a successful mission in a non-ideal environment.

The requirements for this competition are the first steps in developing a complete 3-D mission simulator.

Competition Schedule

- Start 2/11 (Thur) – Kickoff meeting,
- 1 week 2/18 (Thur) – 1 hour Q&A
- 2 weeks 2/25 (Thur) – First demonstration – 1st place will be announced within two days
- 3 weeks 3/4 (Thur) – 1 hour Q&A
- 4 weeks 3/11 (Thur) – Final Demonstration – 1st and 2nd place will be announced within a week

Specifications for Simulator

Requirements:

- Must use MOGRE (it’s easy) as the rendering engine.
  - www.ogre3d.org/wiki/index.php/MOGRE start here
  - www.ogre3d.org/wiki has great tutorials on rendering with OGRE
  - Import submarine specific blender models into OGRE. (These will be provided on the competition website in both blender and OGRE mesh files.)
- Must use C# as the language.
  - Good primer on going from Java to C# http://www.25hoursaday.com/CsharpVsJava.html
- Dynamic obstacle placement into 3-D world. (i.e., red ball (simulated buoy), orange bar (simulated pipe)
- Must design architecture for plugging in new simulated sensors
  - Must implement camera and accelerometer with real time data display
- Must take inputs from keyboard and move submarine within world
  - Must be implemented as a plug-in
  - X-Box controller plug-in can be used … if you think you’re cool enough
- Simulation of Dynamics not required

1st Demo Requirements (3-D Rendering):

- Render 3D objects in a GUI
  - Render environment (TRANSDEC blender model will be provided)
  - Render moving submarine
  - Render static buoy and pipe
- Must show “output” from onboard camera
- Must be able to move submarine via input plug-in (i.e., keyboard, X-Box controller)
- Must be able to use mouse to manipulate third person camera

3-D Simulator Programming Competition

Final Demo Requirements:

- Must render 3-D environment with all obstacles in polished GUI
- Dynamic world building
- Must show fully functioning plug-in framework

What we will provide for you on the competition website:

- Blender Model of Sub
- Blender Model of Buoy and Pipe
- Blender model of environment (TRANSDEC)
- Ogre provides a tool to export blender files to Ogre
- Support via email and two discussion/Q and A sessions

If you plan to attend the kickoff meeting, please RSVP to UF.3D.Sim@gmail.com by 2/10 (Wed)

Email any questions to UF.3D.Sim@gmail.com