Drone WebGL Demo

Davide Bergamaschi

2020

Project specification

- Drone that flies around in a 3D landscape

Project specification

- Drone that flies around in a 3D landscape
- Third-person point of view

Project specification

- Drone that flies around in a 3D landscape
- Third-person point of view
- Drone cannot fly underneath the terrain

Project specification

- Drone that flies around in a 3D landscape
- Third-person point of view
- Drone cannot fly underneath the terrain
- Shading computation done in World-Space coordinates

Implementation features

- Scan-line rendering + Phong shading (World Space)

Implementation features

- Scan-line rendering + Phong shading (World Space)
  - Main direct light source + ambient component

Implementation features

- Scan-line rendering + Phong shading (World Space)
  - Main direct light source + ambient component
  - Lambert diffuse + Phong specular BRDF

Implementation features

- Scan-line rendering + Phong shading (World Space)
  - Main direct light source + ambient component
  - Lambert diffuse + Phong specular BRDF
- Third-person "Look-At" view, with camera following drone

Implementation features

- Scan-line rendering + Phong shading (World Space)
  - Main direct light source + ambient component
  - Lambert diffuse + Phong specular BRDF
- Third-person "Look-At" view, with camera following drone
- User interaction through basic flight control

Implementation features

- Custom artist-made drone model and terrain textures

Implementation features

- Custom artist-made drone model and terrain textures
- Procedural morphology generated on the fly

Implementation features

- Custom artist-made drone model and terrain textures
- Procedural morphology generated on the fly
  - Planar grid of vertices “disturbed” with 2D simplex GPU noise

Implementation features

- Custom artist-made drone model and terrain textures
- Procedural morphology generated on the fly
  - Planar grid of vertices "disturbed" with 2D simplex GPU noise
  - Vertical displacement computed *una tantum* with vertex shader

Implementation features

- Custom artist-made drone model and terrain textures
- Procedural morphology generated on the fly
  - Planar grid of vertices "disturbed" with 2D simplex GPU noise
  - Vertical displacement computed *una tantum* with vertex shader
  - Normals and tangents obtained from analytical derivatives

Implementation features

- Custom artist-made drone model and terrain textures
- Procedural morphology generated on the fly
  - Planar grid of vertices "disturbed" with 2D simplex GPU noise
  - Vertical displacement computed *una tantum* with vertex shader
  - Normals and tangents obtained from analytical derivatives
  - All data is saved to GPU buffers through Transform Feedback

Implementation features

- Custom artist-made drone model and terrain textures
- Procedural morphology generated on the fly
  - Planar grid of vertices "disturbed" with 2D simplex GPU noise
  - Vertical displacement computed *una tantum* with vertex shader
  - Normals and tangents obtained from analytical derivatives
  - All data is saved to GPU buffers through Transform Feedback
- Terrain bump mapping using texture data

Implementation features

- Invisible walls

Implementation features

- Invisible walls
  - Prevent intersection with terrain

Implementation features

- Invisible walls
  - Prevent intersection with terrain
  - Delimit simulation area

Implementation features

- Invisible walls
  - Prevent intersection with terrain
  - Delimit simulation area
- Independently animated propellers

Implementation features

- Invisible walls
  - Prevent intersection with terrain
  - Delimit simulation area
- Independently animated propellers
  - Motion blur effect baked in static texture

Implementation features

- Invisible walls
  - Prevent intersection with terrain
  - Delimit simulation area
- Independently animated propellers
  - Motion blur effect baked in static texture
  - Rotation at constant speed

Implementation features

- Invisible walls
  - Prevent intersection with terrain
  - Delimit simulation area
- Independently animated propellers
  - Motion blur effect baked in static texture
  - Rotation at constant speed
- Exponential distance fog