

Soapbox car

- **Name:** Dylan Van Assche
- **Date:** 24 May 2018
- **Repo:** <https://www.github.com/DylanVanAssche/opengl-car>

A futuristic soapbox car in OpenGL for the course Computergraphics by Herman Crauwels (Campus De Nayer, KU Leuven). This application provides a demo for several OpenGL functions and is written in ANSI C.

Features

- [x] Basic car
- [x] Basic finish
- [x] Multiple 3D transformations (orthogonal, symmetric, ...)
- [x] Multiple cars on the screen
- [x] Show axes, light sources, ... controlled with a keyboard key
- [x] Improved finish
- [x] 4 different light sources
- [x] Switch between several shade models
- [x] Menu to configure the materials used to create the car
- [x] Keyboard shortcuts
- [x] Animates the wheels of the car
- [x] Texture support of the finish and the car
- [x] Toggle the alpha value of the car
- [x] Fog support

Functions

See the inline comments in the source files for more information about the arguments of each function.

car.c

- `void menu(GLint id):` Main menu callback to dispatch several other menus and handling the quit option.
- `void coachworkMenu(GLint id):` Coachwork menu callback to set the coachwork colors.
- `void suspensionMenu(GLint id):` Suspension menu callback to set the suspension colors.
- `void finishMenu(GLint id):` Finish menu callback to set the finish colors.

- `void init(void)`: Init function to initialize several things before the OpenGL main loop has been started.
- `void animation(GLint value)`: Callback for `glutTimerFunc` to animate the soapbox car.
- `void keyboardWatcher(unsigned char key, int x, int y)`: Keyboard callback
- `void displayFunction(void)`: Display callback
- `void windowFunction(GLint newWidth, GLint newHeight)`: Window callback
- `int main(int argc, char* argv[])`: Main function

`finish.c`

- `void _drawFinishPart(GLint wireFrame)`: Private function to draw 1/4 of the finish arc.
- `void drawFinish(GLint wireFrame, GLfloat* ambient, GLfloat* diffuse, GLfloat* specular, GLuint textureAddressing[], GLint texture, GLint checkpoints)`: Draws the complete finish arc.
- `void drawCoachwork(GLint wireFrame, GLfloat* ambient, GLfloat* diffuse, GLfloat* specular, GLint clear, GLint checkpoints, GLint texture)`: Draws the coachwork.

`vehicle.c`

- `void drawSuspension(GLint wireFrame, GLfloat* ambient, GLfloat* diffuse, GLfloat* specular)`: Draws the complete suspension.
- `void drawTires(GLint wireFrame, GLfloat animationAngle, GLuint textureAddressing[], GLint texture)`: Draws the tires of the soapbox car.

`view.c`

- `void drawAxes(GLint axes)`: Draws the X,Y and Z-axis.
- `void drawCheckpoint(const GLfloat* color)`: Draws a single checkpoint, used by the complex curves and light positions.
- `void configureLights(GLint ambientLight, GLint diffuseLight, GLint specularLight, GLint spotLight, GLint spotAngle, GLint spotExponent, GLint spotHeight, GLint drawPositions)`: Configures the lights on the right position with the right colors.
- `void configureFog(GLint fog, GLint fogMode, GLfloat far)`: Configures the fog and it's mode.

Shortcuts

Mouse

- Right click to show the menu where you can select several materials for each component of the car.
- You can also quit the application from the menu if you like.

Keyboard

Key	Description
x/X	Move camera (X axis +/-)
y/Y	Move camera (Y axis +/-)
z/Z	Move camera (Z axis +/-)
i	Move camera to it's initial position
1	Toggle ambient light (GL_LIGHT0)
2	Toggle diffuse light (GL_LIGHT1)
3	Toggle specular light (GL_LIGHT2)
4	Toggle spot light (GL_LIGHT3)
s	Switch to shading mode <i>SMOOTH</i>
S	Switch to shading mode <i>FLAT</i>
l	Turn wireframe mode <i>ON</i>
L	Turn wireframe mode <i>OFF</i>
j	Turn axes <i>ON</i>
J	Turn axes <i>OFF</i>
p	Turn light positions <i>ON</i>
P	Turn light positions <i>OFF</i>
k	Turn checkpoints <i>ON</i>
K	Turn checkpoints <i>OFF</i>
g	Toggle wheel animation
G	Toggle car animation
t	Toggle textures
m	Toggle fog
M	Toggle fog mode (<i>LINEAR</i> / <i>EXP</i>)
n	Toggle competition mode
f	Toggles coachwork transparency
h/H	Moves the spot height (+/-)
v/V	Manipulates the spot angle (+/-)
w/W	Manipulates the spot exponent (+/-)
b	Toggles light locking
e/E	Manipulates the material shininess (+/-)
q/Q	Quits the application

How to build

1. Clone this repo: `git clone https://github.com/DylanVanAssche/opengl-car.git`
2. Enter the directory: `cd opengl-car`
3. Install all OpenGL libraries via your package manager, you can find the dependencies in the MakeFile.
4. Run `make` in the project directory.
5. Run `./car` to start the soapbox car.

License

- Everything (except for the images, `initJPG.c` and `initJPG.h`) in this repository is available under the GPLv3 license.
- The images are available under separate licenses, see the folder `images` for more information.
- For `initJPG.c` and `initJPG.h` files, please check the comments in the files for more information.