

Graphics programming

Exercise 9

Henrique Debarba

IT University of Copenhagen

Exercise 9

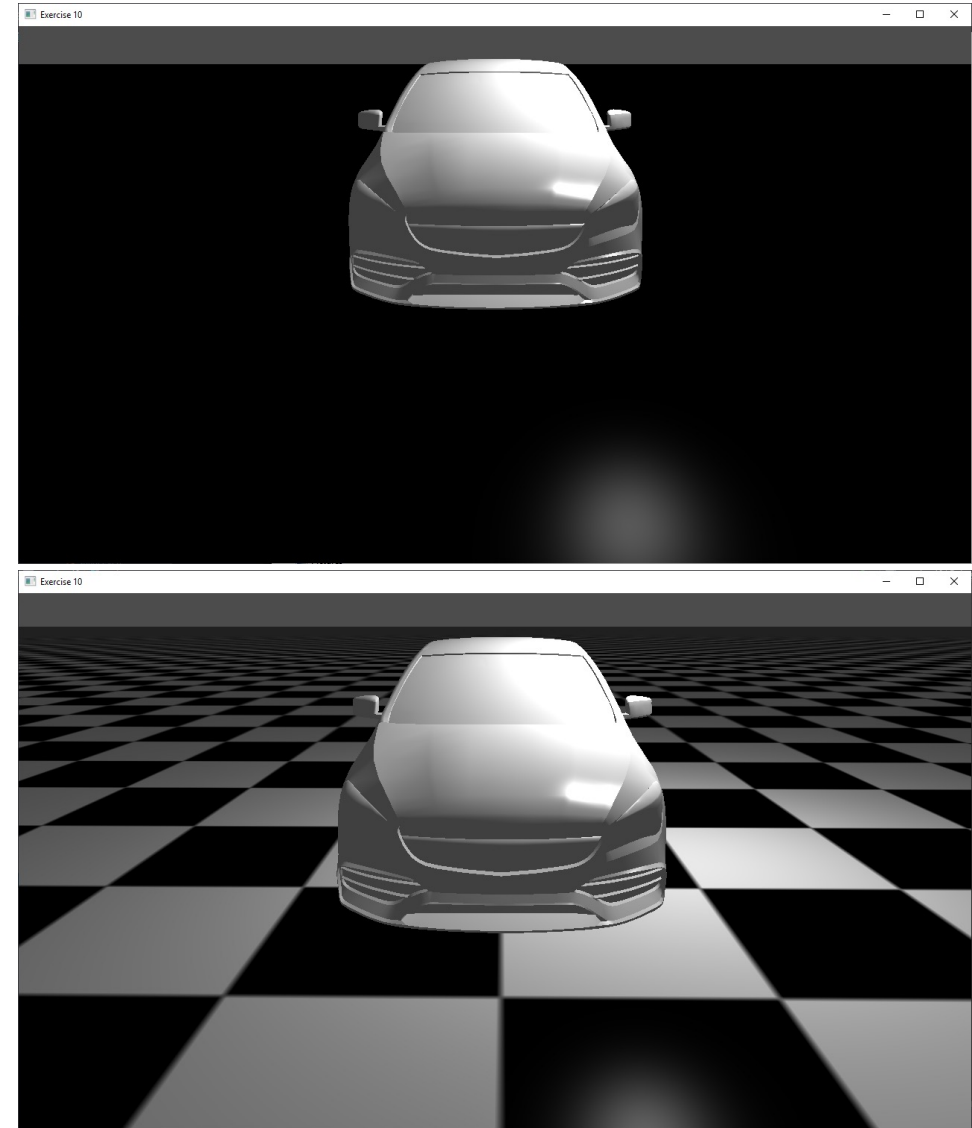
- Learning objectives
 - Load and use a texture in OpenGL and GLSL
 - Experiment with different texture sampling settings
 - Improve lighting based on precomputed information

Exercise 8

- Additional resources
 - <https://learnopengl.com/Getting-started/Textures>
 - <https://learnopengl.com/Lighting/Materials>
 - <https://learnopengl.com/Lighting/Lighting-maps>

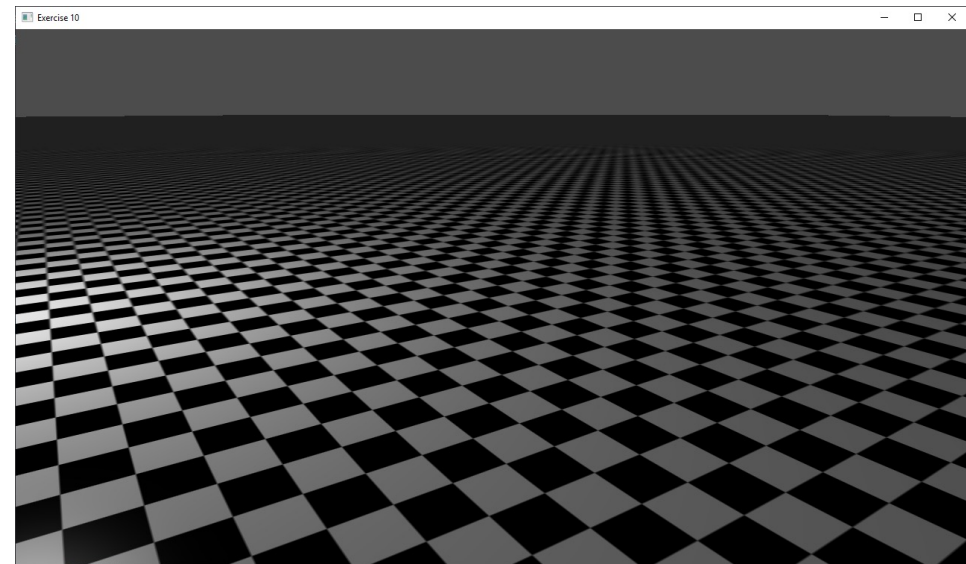
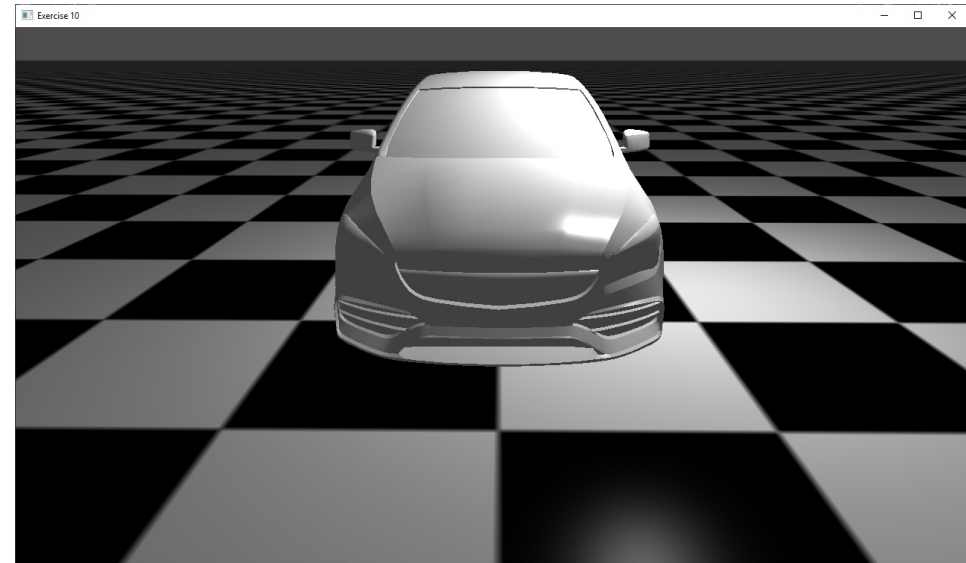
Exercise 9.1

- Loading a texture
 - Based on the **TextureFromFile** function in the **Model.h** file, implement the **loadFloorTexture** in the **main.cpp**.
 - Take the **config.wrapSetting**, **config.minFilterSetting** and **config.magFilterSetting** settings into account.
 - Run the program and visualize the different texture wrap and filtering options.
 - There is a practical overview of the options in <https://learnopengl.com/Getting-started/Textures>



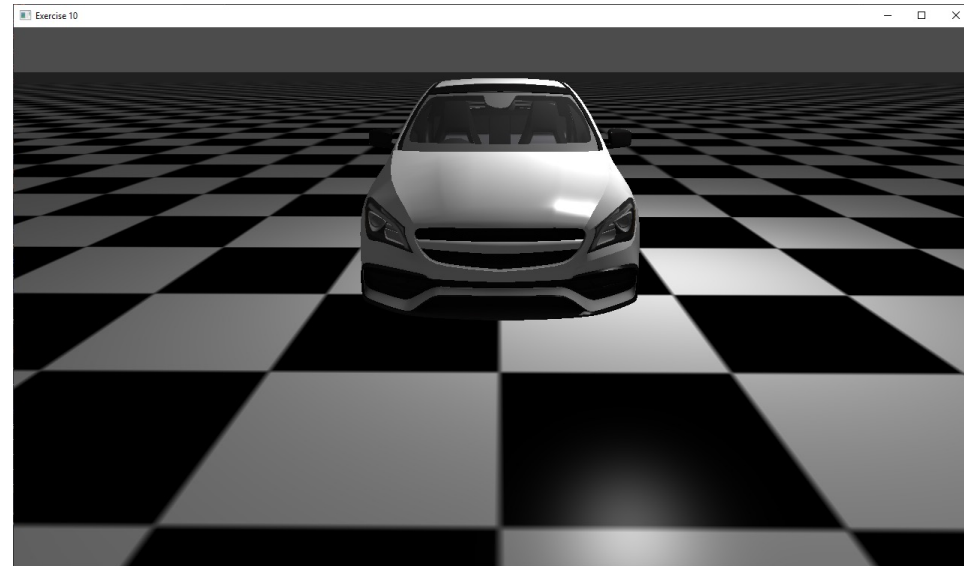
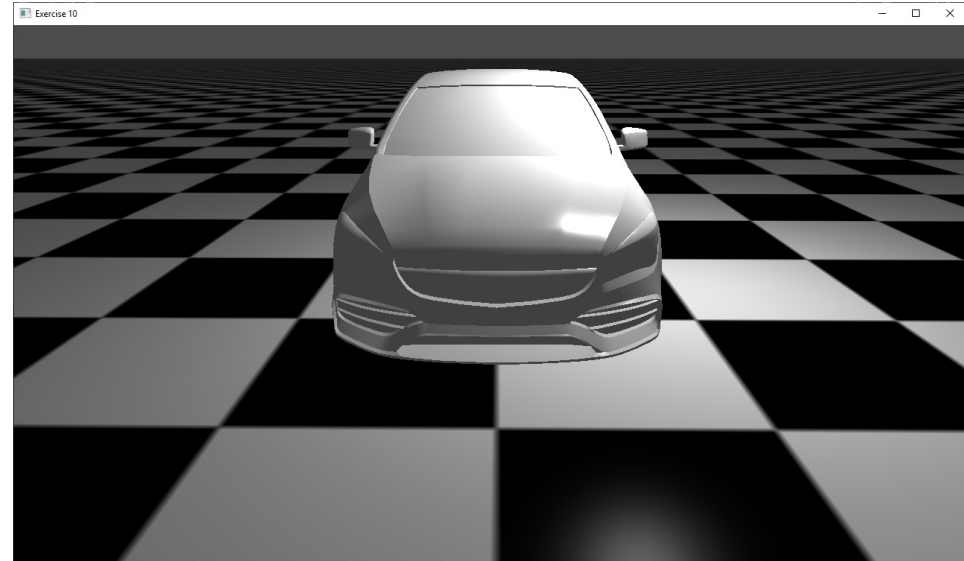
Exercise 9.2

- Scaling uv mapping
 - Have a look in the **common/models/floor/floor_no_material.obj** file. You will see that the texture coordinates (**uvCoords**) are being scaled. This is done so we repeat the same texture several times.
 - Add an **uvScale** to the config struct, a **slider to control uvScale** in the GUI (**drawGui** function), and send it to **floor_shader.frag** shader as a uniform. Use it to scale the **textCoord** it in real time.



Exercise 9.3

- Diffuse textures
 - In the fragment shader, sample the diffuse texture and use it as the **modelColor**.
 - Note that the diffuse texture has 4 channels: **red, green blue and alpha (rgba)**. We don't compute lighting for a.
 - Use the alpha (a) to set the opacity of the fragment.



Exercise 9.4

- Ambient occlusion
 - Sample the **ambient_occlusion1** texture in the fragment shader.
 - Use the **ambientOcclusionMix** setting and the mix function to interpolate between 1.0 and the value sampled in the texture
 - modulate the lighting of the car using the **red** channel (the texture is rgba, but the rgb channels are replicating the data).



Exercise 9.5

- Normal map
 - Next week we will implement a bump mapping technique. It uses surface information stored in an image (normal map) to implement detailed lighting while keeping a low polygon count.
 - You can sample **texture_normal1** instead of **texture_diffuse1** to see what a normal map looks like

