UNIVERSITÄT DES SAARLANDES
Prof. Dr.-Ing. Philipp Slusallek
Computer Graphics Group
Stefan Lemme (lemme@cg.uni-saarland.de)

24. January 2019

Introduction to Computer Graphics
Assignment 9

Submission deadline for the exercises: 31. January 2019

The paper copies for the theoretical parts of the assignments will be collected at the beginning of the lecture on the due date. The programming parts must instead be marked as release before the beginning of the lecture on the due date.
The code submitted for the programming part of the assignments is required to reproduce the provided reference images. The submission ought to tag the respective commit in your group's git repository as well as attach the mandatory generated images to the release. The submission should also contain a creative image show-casing all extra-credit features that have been implemented.
The projects are expected to compile and work out of the box. A successful build by Drone CI is a good indicator. If it fails to do so on our Windows or Linux computers, we will try to do so on the CIP-pool students' lab as a "fallback".
To pass the course you need for every assignment at least 50% of the points.

9.1 Normalization of the Phong BRDF (10 + 20 Points)

The *Phong* BRDF has the following form:

$$f(\omega_i, \omega_o) = k_d + k_s\, cos^n(\omega_r, \omega_o)$$

where ω_r is the result of the reflection of the incoming direction ω_i along the surface normal N. The goal of this exercise is to find the possible values of k_d and k_s so that f is *energy conserving*. In order to do this, we define ρ_s and ρ_d such that:

$$\begin{aligned} k_d &= \frac{\rho_d}{C_d} \\ k_s &= \frac{\rho_s}{C_s} \\ \rho_d + \rho_s &\leq 1 \\ \rho_s, \rho_d &\geq 0 \end{aligned}$$

C_s and C_d are the normalization factors needed to make the BRDF energy conserving.

a) Assume that $k_s = 0$ (this means $\rho_s = 0$ and $\rho_d \in [0, 1]$). Derive the expression of C_d so that f is energy conserving.

b) Assume that $k_d = 0$ (this means $\rho_d = 0$ and $\rho_s \in [0, 1]$). Derive the expression of C_s so that f is energy conserving.
Hint: we are interested in the maximum of the reflected energy, which happens when the outgoing direction is parallel to the normal ($\omega_o = N$). The cosine term becomes $cos(\omega_r, \omega_o) = cos(\omega_r, N) = cos(\omega_i, N)$.

9.2 Camera Transformation (15 Points)

Deduce the complete camera transformation from a left handed coordinate system to a canonical viewing frustum $[-1 \dots 1]^2 \times [0..1]$. Please note that in the lecture a right handed coordinate system was used. It suffices to write down the matrices $(T, R, H, S_{xy}, S_{far}, P_{persp})$. The projection should be down to the xy-plane, along the z-axis. The parameters for the camera are:

- Point of projection: $(2, 3, 1)^T$
- Normal of projection plane: $(1, 1, 1)^T$ (needs to be normalized!)
- Up vector: $(0, 0, -1)^T$
- Focal length: 2
- Clipping window: left: 0, right: 2, bottom: 1, top: 3
- Clipping planes: near: 1, far: 2.5

9.3 Clipping (15 Points)

Given an axis aligned clipping window spanned by the two points $(0, 0)$ and $(12, 6)$

$$C_1 = (0, 0), C_2 = (0, 6), C_3 = (12, 6), C_4 = (12, 0)$$

and a polygon with the lines $P_1 - P_2 - P_3 - P_4 - P_1$ with

$$P_1 = (9, 8), P_2 = (-6, 3), P_3 = (14, 20), P_4 = (15, -4).$$

Use the Sutherland-Hodgeman algorithm to perform the first step of the algorithm by clipping the polygon against the top edge of the clipping window only. Describe each step of the algorithm. You do not have to compute the intersection point of the lines with the clipping window, give them names in a figure and use that names instead.