

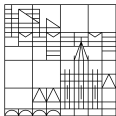
# Global Illumination Methods

Practical Course

25 November 2019

Till Niese

Universität  
Konstanz



# Work Package II

## Tasks

1. Implement Shading-Model: Ambient, Diffuse, Blinn-Phong
2. Procedural texturing
3. Octree implementation (suggested, but optional)

## Date

This assignment is due **December, 9th**. Please bring your Laptop to class. If you have any questions regarding the assignment, just write us an email.

# Task 1

## Ambient shading

- ▶ Shading does not depend on anything
- ▶  $k_a$ : ambient color
- ▶  $I_a$ : ambient light intensity

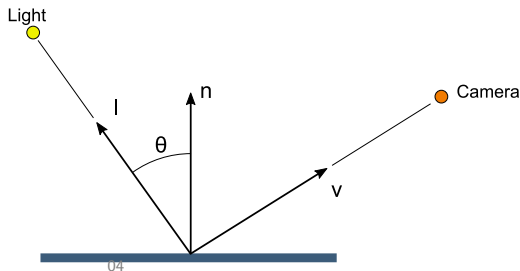
$$L_a = k_a I_a$$

# Task 1

## Diffuse shading

- ▶ Place a light source in the scene.
- ▶ Calculate the surface normal at the hit point.
- ▶ Diffuse shading (without specular highlight) using lambertian shading.
- ▶  $k_d$ : diffuse surface color
- ▶  $I$ : light intensity

$$L_d = k_d I \max(0, n \cdot l)$$



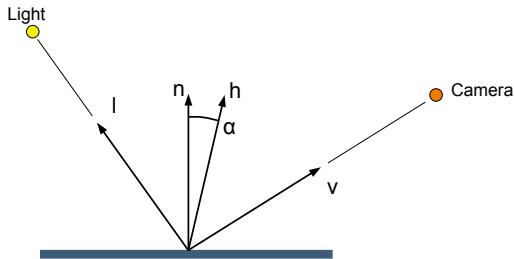
# Task 1

## Specular shading (Blinn-Phong)

- ▶ Place a light source in the scene.
- ▶ Calculate the surface normal at the hit point.
- ▶ Calculate the specular intensity using the using the angle between *bisector* and *surfacenormal*

$$h = \text{bisector}(v, l)$$

$$L_s = k_s I \max(0, n \cdot h)^p$$



## Task 2

### Procedural texturing

Create a checkerboard texture and apply it to a plane and sphere.

## Task 3

### Octree

To improve rendering performance for a large number of objects and triangles.

```
/// Store an entity in the correct position of the octree.  
void push_back(Entity* object);  
  
/// Returns list of entities that have  
/// the possibility to be intersected by the ray.  
std::vector<Entity*> intersect(const Ray& ray) const;  
  
/// Subdivides the current node into 8 children.  
void Node::partition();
```