

4. Übungsblatt zur Vorlesung Computergraphik im WS 2025/26

Besprechung am Montag, Dienstag, und Mittwoch, 01.12., 02.12 & 03.12.2025

Aufgabe 1 *Beleuchtung* [4 Votierpunkte]

Wo zutreffend, nutzen Sie auch die Formel der Phong-Beleuchtung mit den einzelnen Komponenten in Ihren Erklärungen.

1. Erklären Sie den Unterschied zwischen spekularer und diffuser Reflexion. Was bewirkt eine Erhöhung der “Shininess” in der Phong-Beleuchtung?
2. Welche Bedeutung hat die ambiente Komponente bei der Phong-Beleuchtung? Kommt diese in der “realen Welt” vor?
3. Welche der drei Material-Komponenten der Phong-Beleuchtung (diffus, spekulär, ambient) sind abhängig von der Blickposition (viewpoint) des Betrachters, bzw. der Kameraposition? Begründen Sie ihre Antwort für jede der Komponenten.
4. Ein Lichtstrahl trifft von oben auf eine flache horizontale Wasseroberfläche (Temperatur: 20°C) mit einem Winkel von 30° relativ zum Oberflächennormalenvektor auf. Über der Oberfläche ist Luft. Mit welchem Winkel relativ zum Oberflächennormalenvektor wandert der Strahl nach unten im Wasser? Erklären Sie die Rechenschritte.

Aufgabe 2 *Beleuchtung im Raytracer* [6 Votierpunkte]

Eine aktualisierte Version des Raytracers aus dem letzten Übungsblatt steht Ihnen als Skelett zur Verfügung. Das Skelett enthält nun zusätzlich die Klasse `Pointlight`, die Punktlichtquellen repräsentiert. In der Szene sind bereits 16 Punktlichter verteilt. Die `Ray`-Klasse wurde um die Member-Variable `depth` ergänzt, die die rekursive Tiefe des Strahls speichert.

Hinweis: Im *Debug*-Modus kann die Ausführungszeit des Programms mit Schatten- und Reflexionsstrahlen sehr lange werden. Falls Sie Ihre Implementierung im *Debug*-Modus testen, reduzieren Sie am besten vorübergehend die Anzahl der Lichtquellen in der Szene oder die Auflösung des Ausgabebildes.

1. Implementieren Sie in der Datei `vec3.h` die Funktion `reflect` (*TODO 1*), die einen einfallenden Vektor reflektiert (siehe auch Vorlesungsfolien).
2. Implementieren Sie in der Datei `main.cpp` die Funktion `computePhongLighting` (*TODO 2*). Die Funktion soll das Phong-Beleuchtungsmodell berechnen (siehe auch Vorlesungsfolien). Verwenden Sie dabei die Lichtfarbe nur bei der Berechnung des spekularen Anteils.

3. Implementieren Sie in der Datei *main.cpp*, in der Funktion `castRay`, die lokale Beleuchtung für getroffene Szeneobjekte (*TODO 3*). Iterieren Sie dafür über alle Lichtquellen. Konstruieren Sie jeweils einen Schattenstrahl, um zu prüfen, ob der Schnittpunkt mit dem Objekt von der Lichtquelle beleuchtet wird. Berechnen Sie gegebenenfalls die korrekten Eingabeparameter für die Funktion `computePhongLighting` und rufen Sie die Funktion auf, um die lokale Beleuchtung zu berechnen. Falls der Schnittpunkt im Schatten liegt, nutzen Sie zur Beleuchtung nur den ambienten Term des Phong-Beleuchtungsmodells (dafür benötigen Sie keine extra Funktion).
4. Implementieren Sie in der Datei *main.cpp*, in der Funktion `castRay`, die spiegelnde Reflexion für das getroffene Szeneobjekt (*TODO 4*). Konstruieren Sie dafür einen Reflexionsstrahl und rufen Sie die Funktion `castRay` rekursiv auf. Denken Sie daran, die Rekursionstiefe im Reflexionsstrahl sinnvoll zu setzen, da ihr Programm sonst in eine Endlosschleife gerät!

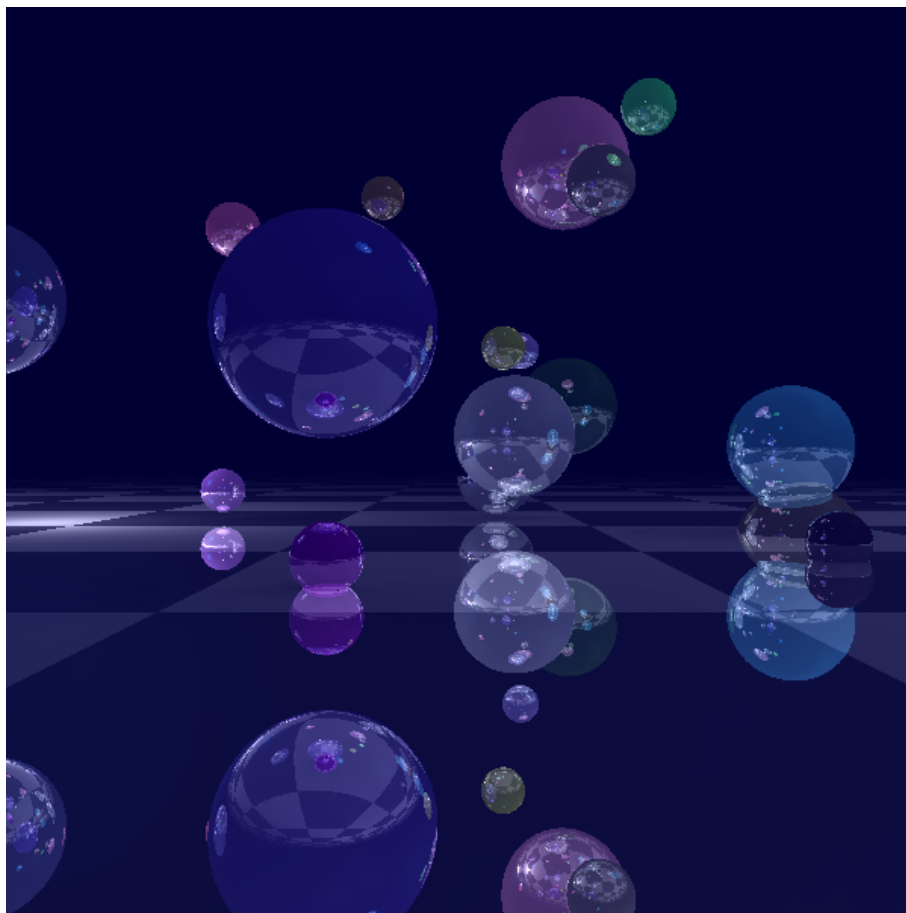


Abbildung 1: Referenzbild der beleuchteten Szene mit Reflexionen bei einer Auflösung von 600x600 Pixeln

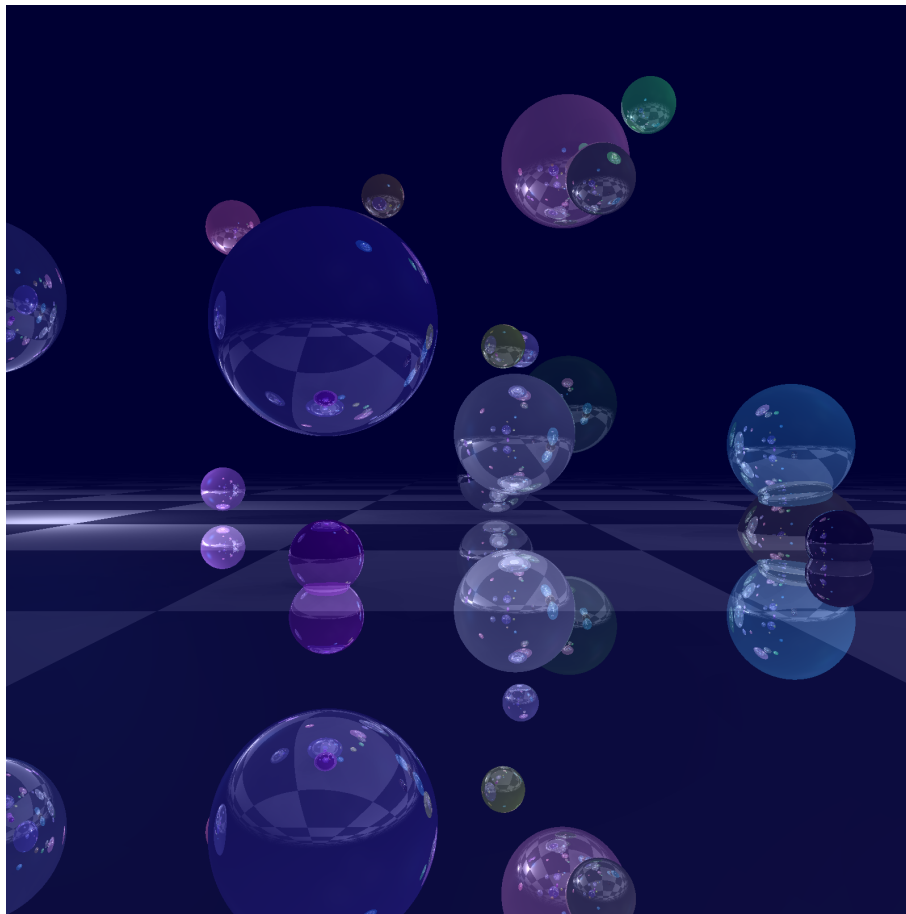


Abbildung 2: Referenzbild der beleuchteten Szene mit Reflexionen bei einer Auflösung von 1600x1600 Pixeln.