Computer Graphics 2024

U3087 / M5222

National Taipei University

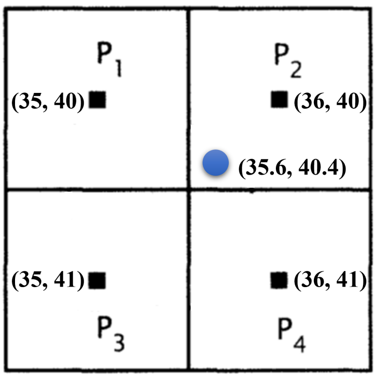
Final Exam

1. What are the advantages of phenomenological models (e.g., Phong lighting model) and geometric optics models (e.g., the microfacet model), respectively? (6%)
   A phenomenological model is easy to control using intuitive parameters but lacks accuracy, while a geometric optics model is challenging for artists to manipulate due to the required physics background.

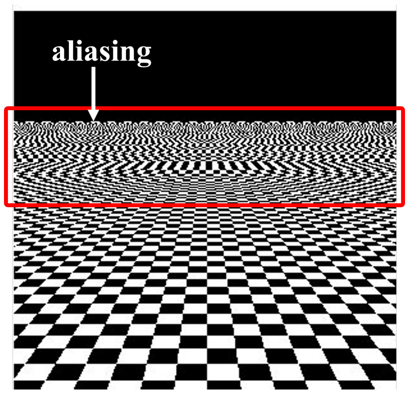
2. Explain the concept of the microfacet model. (4%)
   Rough surfaces can be modeled as a collection of small microfacets, where the aggregate behavior of these small microfacets determines the overall scattering property.

3. Explain how 2D textures can be used to represent spatially varying material parameters on 3D models with arbitrary shapes. (6%)
   A vertex attribute called a texture coordinate is introduced to map a 3D surface position to a 2D location on an image texture. During rendering, the texture coordinate is interpolated for a fragment by rasterization and used to sample the texture, determining the fragment's material properties.

4. For the scenario shown in Fig. 1a, what is the value at coordinate **X** (35.6, 40.4) using **bilinear filtering**? The relevant texture data are as follows: **P1**, centered at (35, 40), has a value of **50**; **P2**, centered at (36, 40), has a value of **20**; **P3**, centered at (35, 41), has a value of **80**; and **P4**, centered at (36, 41), has a value of **100**. (6%)
   The result is $50 * 0.4 * 0.6 + 20 * 0.6 * 0.6 + 80 * 0.4 * 0.4 + 100 * 0.6 * 0.4 = 56$



(a) bilinear filtering



(b) texture aliasing

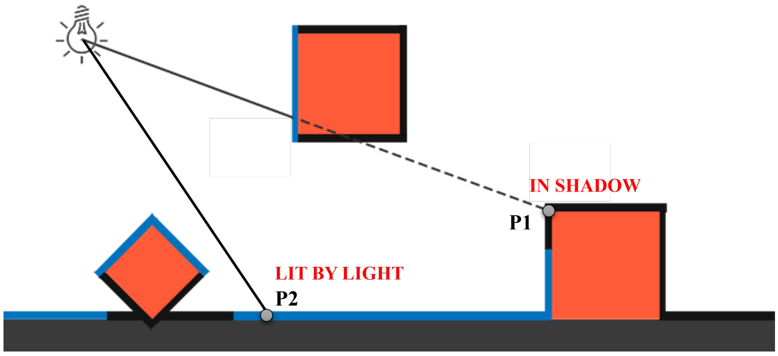
Figure 1: (a) Figure for question 4. (b) Figure for question 5.

5. Answer the following questions about the **mipmap** technique:
   (a) Explain the root cause of texture aliasing observed in Fig. 1b (4%)
   In the labeled area, a single screen pixel corresponds to a surface area larger than one texel in the texture. Consequently, sampling the texture only once results in undersampling and causes aliasing.
   (b) Mipmap is a popular technique to address the issue in (a). Please explain how it works, including the process of constructing a mipmap for a texture and determining the color for texture mapping. (8%)
   The accurate approach is to average the texel colors within the footprint covered by a screen pixel. However, this is too computationally expensive at runtime.
   To reduce the cost, mipmapping pre-filters the texture image during preprocessing, creating a hierarchical representation (image pyramid) where each level has half the resolution of the previous one. During rendering, the screen-space texture coordinates are used to estimate the footprint in the texture space. The two closest levels in the hierarchy are identified, and a trilinear filtering process (bilinear within levels and linear between levels) is applied to compute the final color.

6. Explain how **OpenGL** renders transparent objects. Your answer should discuss the rendering order and explain the blending equation: $\bar{C}_{\text{result}} = \bar{C}_{\text{source}} * F_{\text{source}} + \bar{C}_{\text{destination}} * F_{\text{destination}}$ (8%)
   First, render opaque objects in any order, then render transparent objects in order of their distance from the camera (farthest first), blending each with the color buffer using the blending equation.
   $\bar{C}_{\text{source}}$ and F_{source} represent the color of the current fragment being rendered and its weight, typically determined by its opacity. $\bar{C}_{\text{destination}}$ and $F_{\text{destination}}$ refer to the color buffer's existing value at that fragment and its weight. The result of the blending equation determines the new color value of the fragment in the color buffer.

7. Answer the following questions about **deferred shading**:
   (a) Explain the issue that Deferred Shading is designed to address. (6%)
   If a scene contains many lights and occlusions, significant computational resources are wasted on shading surfaces that are ultimately discarded by Z-buffer tests due to occlusion.
   (b) Following the previous question, explain how **deferred shading** works. (6%)
   Deferred shading employs a two-pass rendering algorithm. The first pass identifies all visible surfaces from the camera by rendering their geometry and material properties in geometry buffers (G-buffers). In the second pass, lighting is computed exclusively for the visible surfaces using the data stored in the G-buffers.

8. Shadow mapping is a two-pass algorithm. Describe the task of each pass and demonstrate how to determine whether **P1** and **P2** are in shadow or not. (10%)
   It uses a two-pass algorithm.
   In the first pass, a depth map of a scene is rendered from the light's perspective to capture the closest surfaces relative to the light source.
   In the second pass, the scene is rendered from the camera's view. At each shading point, shadows are determined by projecting the fragment onto the shadow map and comparing its corresponding values in the map: if a fragment's distance from the light source is greater than the corresponding depth stored in the shadow map (e.g.,



point P1), it is considered to be in shadow. Otherwise, it is illuminated by the light source (e.g., point P2).

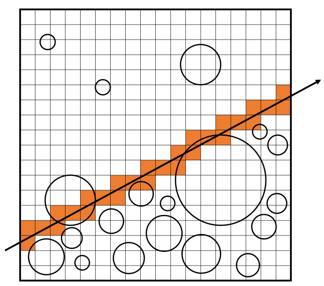
9. Answer the following questions about **ambient lighting** and **ambient occlusion**:

   (a) Explain why **Phong lighting model** includes an ambient term (4%)
   Simulating indirect illumination is computationally expensive. However, without it, any point not directly lit by a light source would appear completely black. The Phong lighting model adds a constant value to approximate indirect illumination to address this.

   (b) **Ambient occlusion** is a widely used technique to enhance the ambient term. Explain how it operates. (4%)
   Ambient occlusion adjusts the ambient term based on the surface's accessibility. Points that are harder to reach, such as those in concave areas, appear darker.

10. Explain the process of rendering an image using **raycasting**. How does **Whitted ray tracing** enhance this method? (10%)
For each pixel, raycasting traces a camera ray through the pixel and finds the closest surface intersection in the scene. The pixel's color is then determined by computing shading at the intersection point.
In addition to camera rays, Whitted ray tracing recursively traces three types of rays: shadow rays, which determine whether a light contributes to a shading point (originating from the shading point toward the light source); reflected rays, which compute perfect specular reflection (originating from the shading point in the direction of the perfect reflection); and refracted rays, which accounts for light refraction (originating from the shading point in the direction dictated by Snell's law).

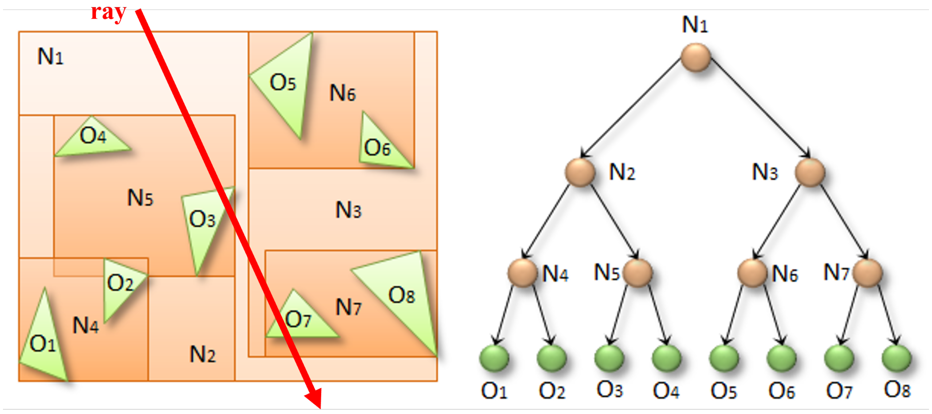
11. Explain why the **uniform grid** is less commonly used as an acceleration structure, despite its simplicity in construction. (6%)



The uniform grid has two main weaknesses. First, determining the optimal grid

resolution is challenging. A coarse grid allows rays to pass through fewer cells but increases the number of intersection tests, as each cell contains more shapes. Conversely, a finer grid reduces the number of intersection tests per cell but increases the time to traverse the ray.
The second weakness arises when the scene has a non-uniform, in-balance distribution of shapes. In regions with many shapes, each grid cell contains numerous shapes, leading to excessive intersection tests. In regions with few shapes, rays traverse many empty cells, reducing efficiency.

12. Using the figure below to explain how a **bounding volume hierarchy (BVH)** enhances the efficiency of ray tracing. Include the traversal sequence in your explanation. (6%)



We first do an intersection test with the scene's bounding box, N1. Since an intersection is detected, we proceed to test its two child nodes, N2 and N3. Both nodes also result in intersections, so we continue testing with their children: N4, N5, N6, and N7. At this level, intersections are found only with N5 and N7. Therefore, we only need to test the primitives O3, O4, O7, and O8.

13. Compare the **strengths** and **weaknesses** of rasterization and ray tracing. (6%)
Rasterization offers greater parallelism and is well-suited for the GPU architecture; however, it struggles to simulate inter-object effects like shadows, indirect lighting, and transparency; In contrast, ray tracing provides a united approach to simulate all light paths. While more versatile, it is computationally expensive due to ray traversal against the complex scene and prone to noise when sample counts are insufficient.