What Is Direct and Indirect Lighting?

In computer graphics, the right balance between direct and indirect lighting elevates the photorealism of a scene.

Author: JJ Kim

Imagine hiking to a lake on a summer day — sitting under a shady tree and watching the water gleam under the sun.

In this scene, the differences between light and shadow are examples of direct and indirect lighting.

The sun shines onto the lake and the trees, making the water look like it's shimmering and the leaves appear bright green. That's direct lighting. And though the trees cast shadows, sunlight still bounces off the ground and other trees, casting light on the shady area around you. That's indirect lighting.

For computer graphics to immerse viewers in photorealistic environments, it's important to accurately simulate the behavior of light to achieve the proper balance of direct and indirect lighting.

Light shining onto an object is called direct lighting .

It determines the color and quantity of light that reaches a surface from a light source, but ignores all light that may arrive at the surface from any other sources, such as after reflection or refraction. Direct lighting also determines the amount of the light that's absorbed and reflected by the surface itself.

Light bouncing off a surface, illuminating other objects is called indirect lighting . It arrives at surfaces from everything except light sources. In other words, indirect lighting determines the color and quantity of all other light that arrives at a surface. Most commonly, indirect light is reflected from one surface onto other surfaces.

Indirect lighting generally tends to be more difficult and expensive to compute than direct lighting. This is because there is a substantially larger number of paths that light can take between the light emitter and the observer.

Global illumination is the process of computing the color and quantity of all light — both direct and indirect — that is on visible surfaces in a scene.

Accurately simulating all types of indirect light is extremely difficult, especially if the scene includes complex materials such as glass, water and shiny metals — or if the scene has scattering in clouds, smoke, fog or other elements known as volumetric media.

As a result, real-time graphics solutions for global illumination are typically limited to computing a subset of the indirect light — commonly for surfaces with diffuse (aka matte) materials.

Many algorithms can be used for computing direct lighting, all of which have strengths and weaknesses. For example, if the scene has a single light and no shadows, direct illumination is trivial to compute, but it won't look very realistic. On the other hand, when a scene has multiple light sources, processing them all for each surface can become expensive.

To tackle these issues, optimized algorithms and shading techniques were developed, such as deferred or clustered shading. These algorithms reduce the number of surface and light interactions to be computed.

Shadows can be added through a number of techniques, including shadow maps, stencil shadow volumes and ray tracing .

Shadow mapping has two steps. First, the scene is rendered from the light's point of view into a special texture called the shadow map. Then, the shadow map is used to test whether surfaces visible on the screen are also visible from the light’s point of view. Shadow maps come with many limitations and artifacts, and quickly become expensive as the number of lights in the scene increases.

Stencil shadow volumes are based on extruding scene geometry away from the light, and rendering that extruded geometry into the stencil buffer. The contents of the stencil buffer are then used to determine if a given surface on the screen is in shadow or not. Stencil shadows are always sharp, unnaturally so, but they don’t suffer from common shadow map problems.

Until the introduction of NVIDIA RTX technology , ray tracing was too costly to use when computing shadows. Ray tracing is a method of rendering in graphics that simulates the physical behavior of light. Tracing the rays from a surface on the screen to a light allows for the computation of shadows, but this can be challenging if the light comes from one point. And ray-traced shadows can quickly get expensive if there are many lights in the scene.

More efficient sampling methods were developed to reduce the number of rays required to compute soft shadows from multiple lights. One example is an algorithm called ReSTIR , which calculates direct lighting from millions of lights and shadows with ray tracing at interactive frame rates.

For indirect lighting and global illumination, even more methods exist. The most straightforward is called path tracing , where random light paths are simulated for each visible surface. Some of these paths reach lights and contribute to the finished scene, while others do not.

Path tracing is the most accurate method capable of producing results that fully represent lighting in a scene, matching the accuracy of mathematical models for materials and lights. Path tracing can be very expensive to compute, but it’s considered the “holy grail” of real-time graphics.

Direct lighting provides the basic appearance of realism, and indirect lighting makes scenes look richer and more natural.

One way indirect lighting has been used in many video games is through omnipresent ambient light. This type of light can be constant, or vary spatially over light probes arranged in a grid pattern. It can also be rendered into a texture that is wrapped around static objects in a scene — this method is known as a “light map.”

In most cases, ambient light is shadowed by a function of geometry around the surface called ambient occlusion, which helps increase the image realism.

Direct and indirect lighting has been present, in some form, in almost every 3D game since the 1990s. Below are some milestones of how lighting has been implemented in popular titles:

1993: Doom showcased one of the first examples of dynamic lighting. The game could vary the light intensity per sector, which made textures lighter or darker, and was used to simulate dim and bright areas or flickering lights.

1995: Quake introduced light maps, which were pre-computed for each level in the game. The light maps could modulate the ambient light intensity.

1997: Quake II added color to the light maps, as well as dynamic lighting from projectiles and explosions.

2001: Silent Hill 2 showcased per-pixel lighting and shadow mapping. Shrek used deferred lighting and stencil shadows.

2007: Crysis showed dynamic screen-space ambient occlusion, which uses pixel depth to give a sense of changes in lighting.

2008: Quake Wars : Ray Traced became the first game tech demo to use ray-traced reflections.

2011: Crysis 2 became the first game to include screen-space reflections, which is a popular technique for reusing screen-space data to calculate reflections.

2016: Rise of the Tomb Raider became the first game to use voxel-based ambient occlusion .

2018: Battlefield V became the first commercial game to use ray-traced reflections.

2019: Q2VKPT became the first game to implement path tracing, which was later refined in Quake II RTX .

2020: Minecraft with RTX used path tracing with RTX.

Real-time graphics are moving toward a more complete simulation of light in scenes with increasing complexity.

ReSTIR dramatically expands the possibilities of artists to use multiple lights in games. Its newer variant, ReSTIR GI , applies the same ideas toward global illumination, enabling path tracing with more bounces and fewer approximations. It can also render less noisy images faster. And more algorithms are being developed to make path tracing faster and more accessible.

Using a complete simulation of lighting effects with ray tracing also means that the rendered images can contain some noise. Clearing that noise, or “denoising,” is another area of active research.

More technologies are being developed to help games effectively denoise lighting in complex, highly detailed scenes with lots of motion at real-time frame rates. This challenge is being approached from two ends: advanced sampling algorithms that generate less noise and advanced denoisers that can handle increasingly difficult situations.

Check out NVIDIA’s solutions for direct lighting and indirect lighting , and access NVIDIA resources for game development .

Learn more about graphics with NVIDIA at SIGGRAPH and watch NVIDIA’s special address , presented by NVIDIA CEO and senior leaders, to hear the latest graphics announcements.

Dive deeper into graphics and rendering on the NVIDIA Technical Blog .

Original URL: https://blogs.nvidia.com/blog/2022/08/04/direct-indirect-lighting/