Motivation

Games on mobile phones, tablets, and laptops have graphics processing limited by heat dissipation in their small form factors. CloudLight is a system for real-time game rendering that computes lighting on distributed servers separate from the processors that compute the final images. This allows even better graphics than today's PC and console games on low-power mobile platforms.

Cloud graphics also offers virtualization, load balancing, lower production costs, and piracy protection for developers. CloudLight offers three different pipelines, targeting different client devices.

Indirect Lighting Pipelines

We developed three different pipelines within CloudLight to solve this for different network and client scenarios:

- **Voxel cone-tracing**: Clients only decode final frames computed on the server and streamed as a video. This allows the most amortization of server resources, but also requires a very low latency network.

- **Irradiance maps**: Clients render direct light and combine it with indirect light streamed asynchronously from server. This is good for medium-power clients with low latency that are limited by low bandwidth and is robust to network instabilities.

- **Photon mapping**: Clients reconstruct indirect light from photons generated on the server. This is gives the most reactive lighting through progressive update for a PC or console-class client.

Distributed System

CloudLight maps the traditional graphics pipeline onto a distributed system which separates direct (view-dependent) and indirect (view-independent) lighting computations so that they can be computed on different machines connected by a network and compressed data streaming.

A distributed graphics pipeline has several differences from a single-machine one. It must:

- Leverage the asymmetric resource distribution between stages of the pipeline.

- Tolerate high latency and low bandwidth between certain pipeline stages.

- Amortize computation across users.

Results

We demonstrate scaling up to 50 users per server node, allowing better quality global illumination for similar per-user costs.

For each technique, we analyse the server-side overhead depending on the number of clients, as well as the effective latency of indirect lighting. We show that only coarse synchronization between direct and indirect light is necessary, and even high latencies can be acceptable for indirect lighting.

For voxel, the common overhead becomes quickly insignificant with increasing number of clients. Each K5000 GPU can simultaneously serve at least 5 clients with a consistent 30 FPS. For irradiance map, per-user latency is essentially constant. Due to larger bandwidth requirements, photon mapping does not scale as well.

Main references:

- Toward practical real-time photon mapping. Efficient gpu density estimation. ACM ISD 2013. MARA, M., LUEBKE, D., AND MCGUIRE, M.
- Shading in Valve's source engine. ACM SIGGRAPH Courses 2006. MITCHELL, J., MC TAGGART, G., AND GREEN, C.