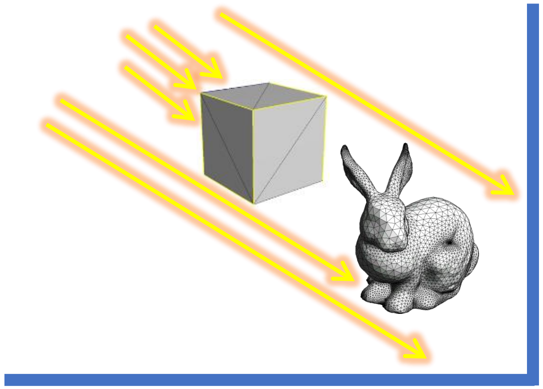
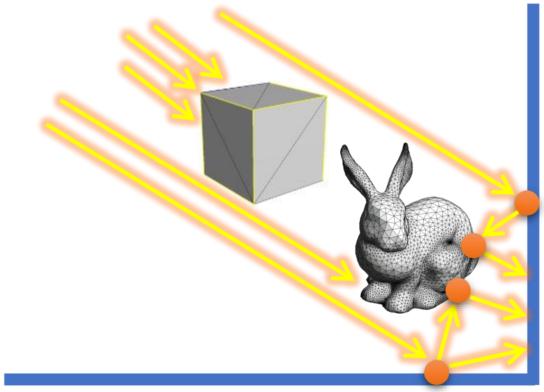
Global Illumination

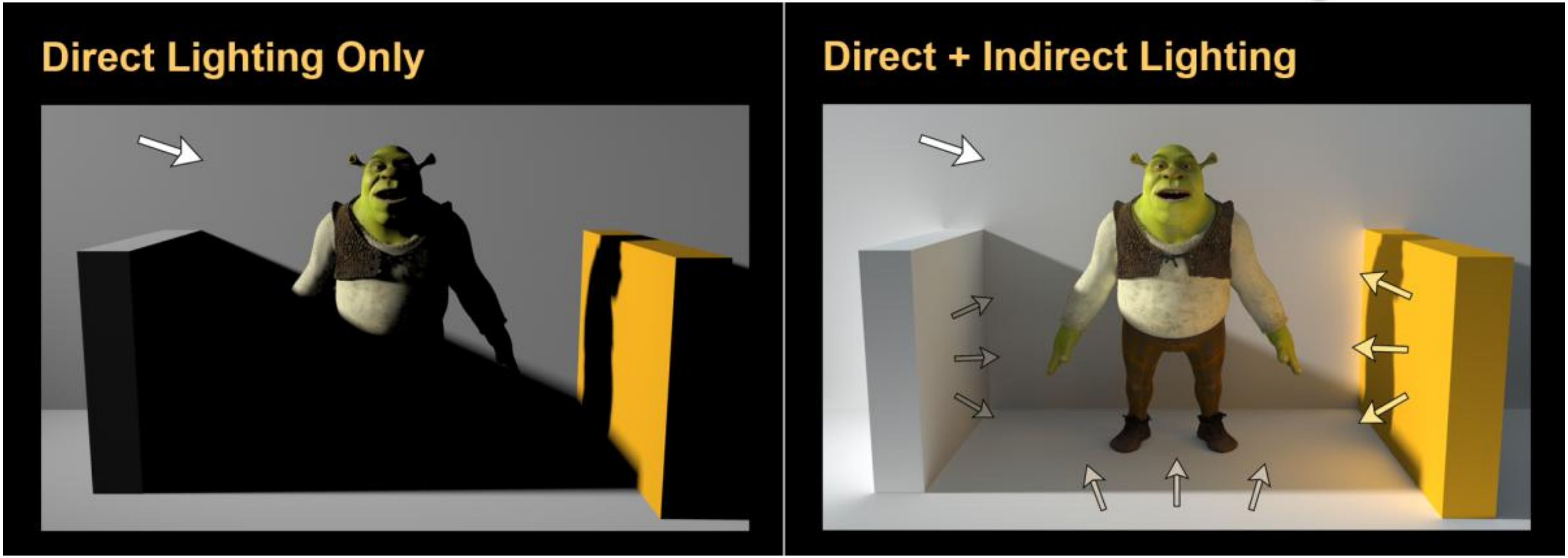
Computer Graphics

Yu-Ting Wu

Global Illumination







Global Illumination (cont.)

global illumination =

direct illumination **+** **indirect illumination**

local illumination + shadow map

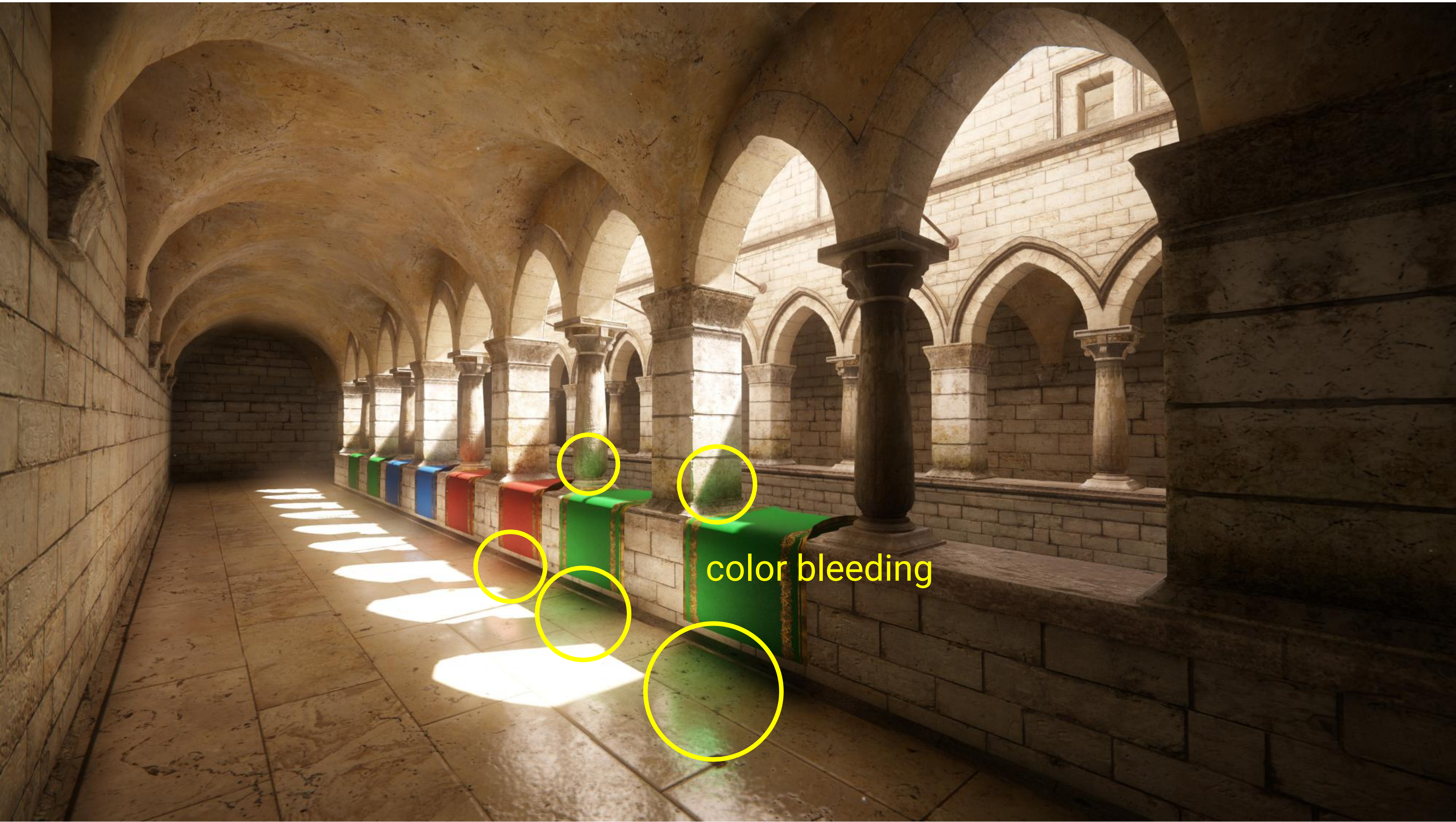
difficult

constant ambient term

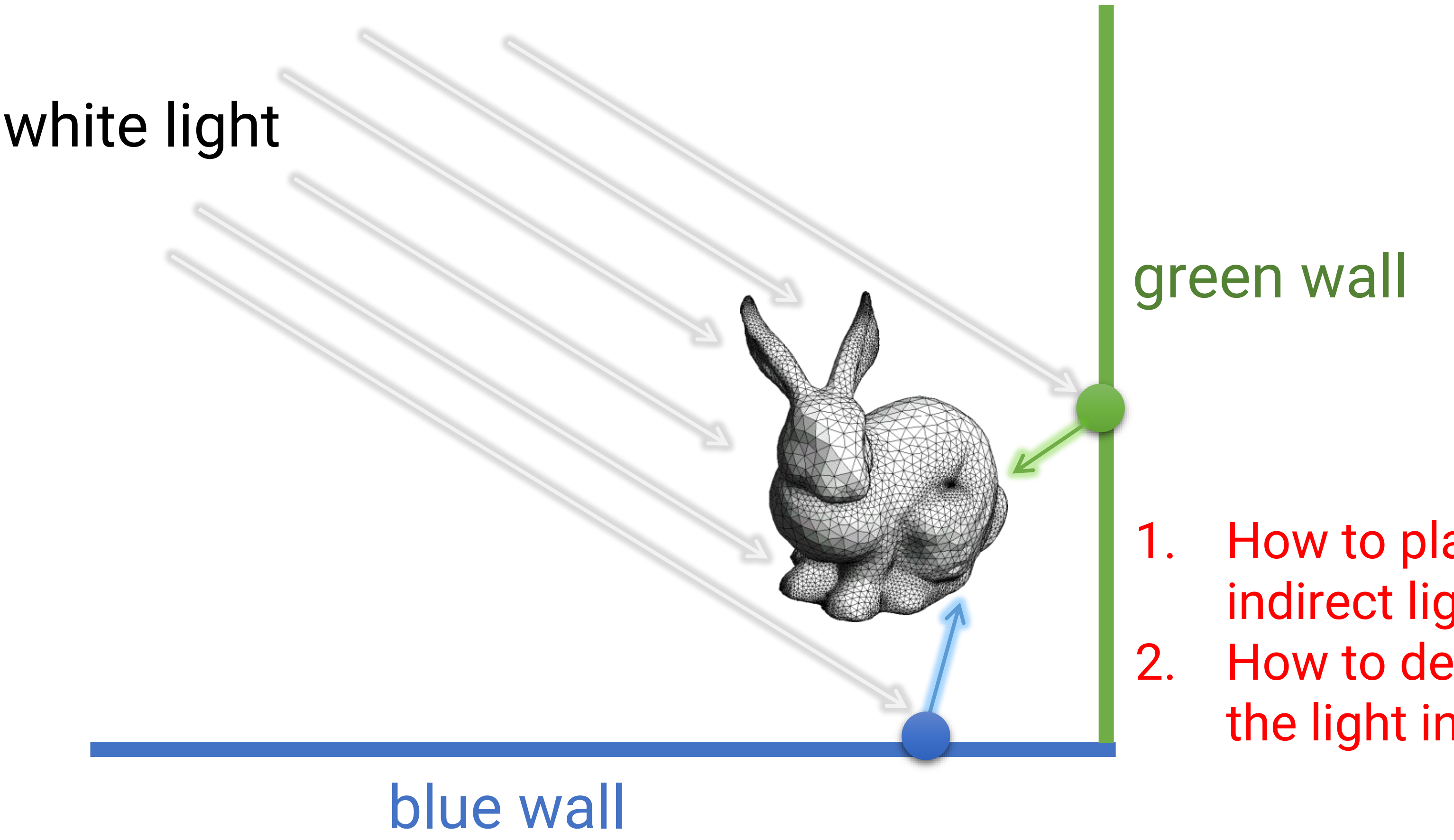
+ ambient occlusion

not good enough

Global Illumination (cont.)



Global Illumination (cont.)



1. How to place the indirect light sources?
2. How to determine the light intensities?

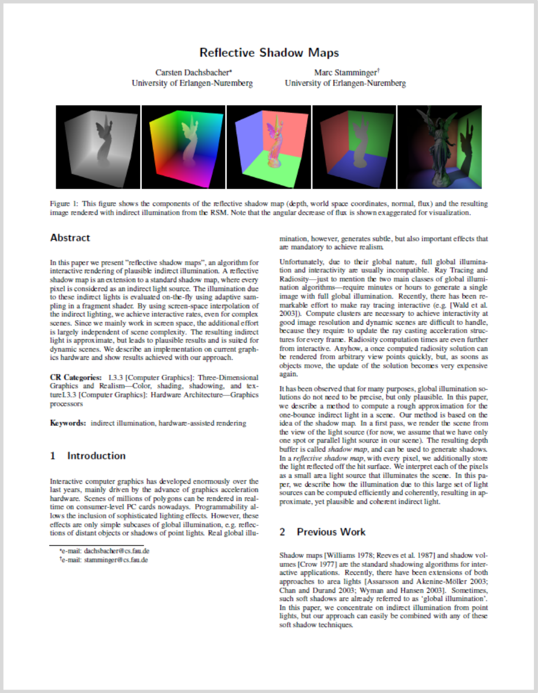
Global Illumination (cont.)

- Indirect illumination is especially difficult for rasterization because …
  - **Each polygon only has its own information**
  - **It does not know which triangle will cast lighting on it**

- In the last two decades, hundreds of research papers focus on this topic to approximate visually-pleasing global illumination in real-time

Reflective Shadow Map

- Proposed by Dachsbacher and Stamminger, I3D 2005
- A classic real-time solution for indirect lighting
- Extend the idea of shadow mapping



Reflective Shadow Maps

Carsten Dachsbacher[*]
University of Erlangen-Nuremberg

Marc Stamminger[†]
University of Erlangen-Nuremberg

Figure 1: This figure shows the components of the reflective shadow map (depth, world space coordinates, normal, flux) and the resulting image rendered with indirect illumination from the RSM. Note that the angular decrease of flux is shown exaggerated for visualization.

Abstract

In this paper we present "reflective shadow maps", an algorithm for interactive rendering of plausible indirect illumination. A reflective shadow map is an extension to a standard shadow map, where every pixel is considered as an indirect light source. The illumination due to these indirect lights is evaluated on-the-fly using adaptive sampling in a fragment shader. By using screen-space interpolation of the indirect lighting, we achieve interactive rates, even for complex scenes. Since we mainly work in screen space, the additional effort is largely independent of scene complexity. The resulting indirect light is approximate, but leads to plausible results and is suited for dynamic scenes. We describe an implementation on current graphics hardware and show results achieved with our approach.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and textureI.3.3 [Computer Graphics]: Hardware Architecture—Graphics processors

Keywords: indirect illumination, hardware-assisted rendering

1 Introduction

Interactive computer graphics has developed enormously over the last years, mainly driven by the advance of graphics acceleration hardware. Scenes of millions of polygons can be rendered in real-time on consumer-level PC cards nowadays. Programmability allows the inclusion of sophisticated lighting effects. However, these effects are only simple subcases of global illumination, e.g. reflections of distant objects or shadows of point lights. Real global illumination, however, generates subtle, but also important effects that are mandatory to achieve realism.

Unfortunately, due to their global nature, full global illumination and interactivity are usually incompatible. Ray Tracing and Radiosity—just to mention the two main classes of global illumination algorithms—require minutes or hours to generate a single image with full global illumination. Recently, there has been remarkable effort to make ray tracing interactive (e.g. [Wald et al. 2003]). Compute clusters are necessary to achieve interactivity at good image resolution and dynamic scenes are difficult to handle, because they require to update the ray casting acceleration structures for every frame. Radiosity computation times are even further from interactive. Anyhow, a once computed radiosity solution can be rendered from arbitrary view points quickly, but, as soons as objects move, the update of the solution becomes very expensive again.

It has been observed that for many purposes, global illumination solutions do not need to be precise, but only plausible. In this paper, we describe a method to compute a rough approximation for the one-bounce indirect light in a scene. Our method is based on the idea of the shadow map. In a first pass, we render the scene from the view of the light source (for now, we assume that we have only one spot or parallel light source in our scene). The resulting depth buffer is called *shadow map*, and can be used to generate shadows. In a *reflective shadow map*, with every pixel, we additionally store the light reflected off the hit surface. We interpret each of the pixels as a small area light source that illuminates the scene. In this paper, we describe how the illumination due to this large set of light sources can be computed efficiently and coherently, resulting in approximate, yet plausible and coherent indirect light.

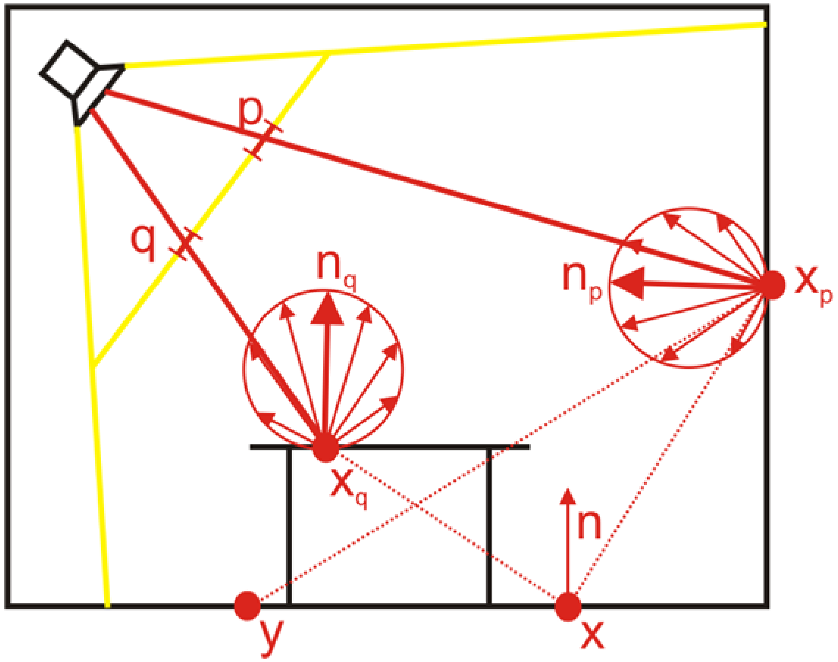
2 Previous Work

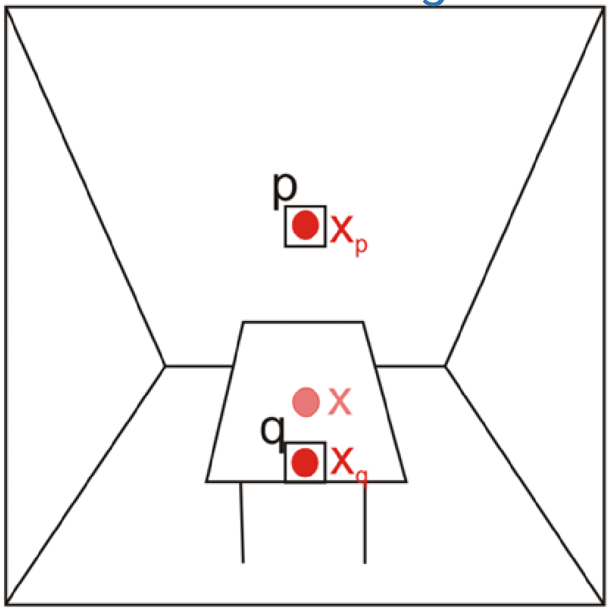
Shadow maps [Williams 1978; Reeves et al. 1987] and shadow volumes [Crow 1977] are the standard shadowing algorithms for interactive applications. Recently, there have been extensions of both approaches to area lights [Assarsson and Akenine-Möller 2003; Chan and Durand 2003; Wyman and Hansen 2003]. Sometimes, such soft shadows are already referred to as 'global illumination'. In this paper, we concentrate on indirect illumination from point lights, but our approach can easily be combined with any of these soft shadow techniques.

[*] e-mail: dachsbacher@cs.fau.de
[†] e-mail: stamminger@cs.fau.de

Reflective Shadow Map (cont.)

- **Major idea**
  - The closest surfaces from the light can receive the lighting contribution
  - They become the indirect light sources

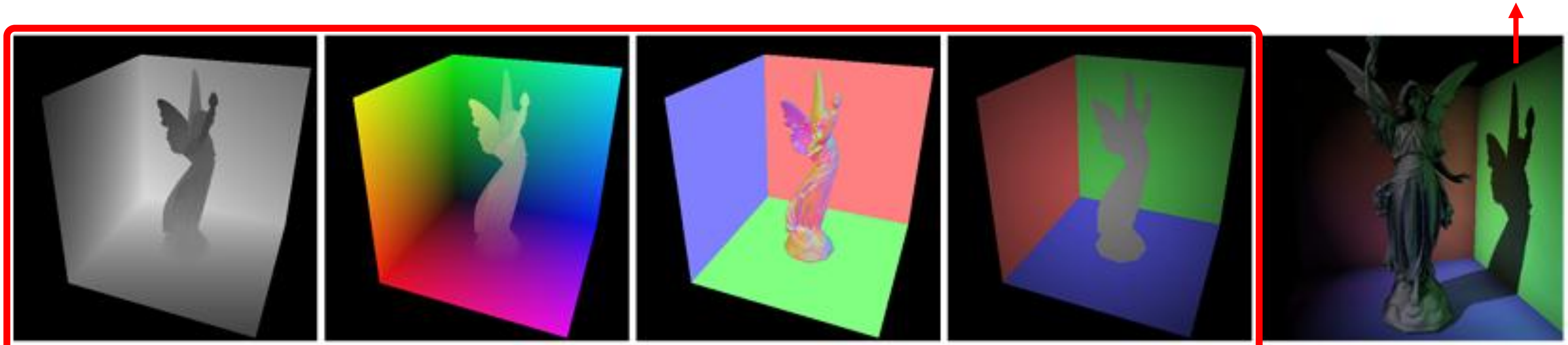




Reflective Shadow Map (cont.)

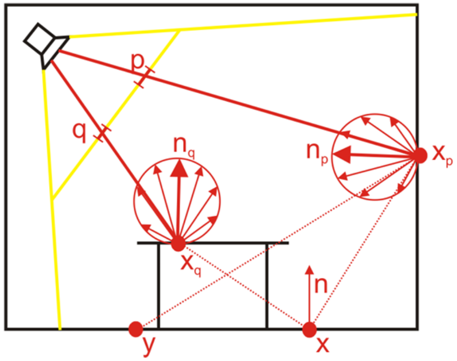
- Two-pass rendering algorithm

Pass II: render from the camera view



depth world position world normal flux

Pass I: render G-buffer from a light view (called RSM)

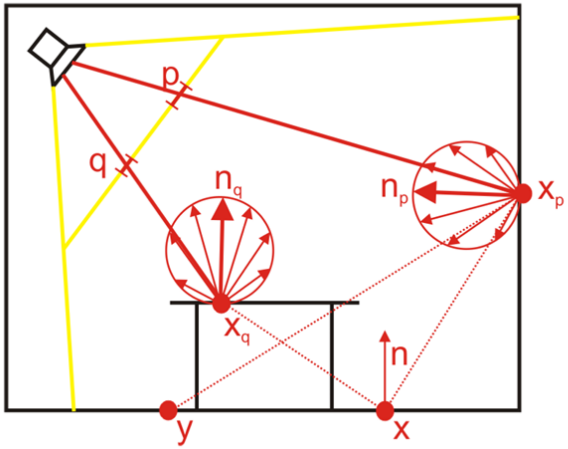


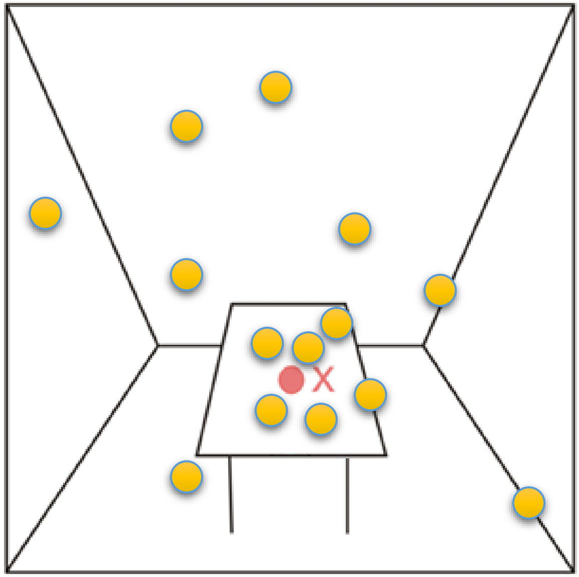
Reflective Shadow Map (cont.)

- **Pass I:** rendering G-buffer (called **RSM**) from the light view for generating indirect light sources
  - World-space position
  - World-space normal
  - Reflected flux
    - The intensity of the primary light source multiplied by the reflectance of the surface
- **Pass II:** rendering from the camera view
  - Direct lighting is computed by local illumination and shadow mapping
  - Indirect lighting is estimated from the RSM

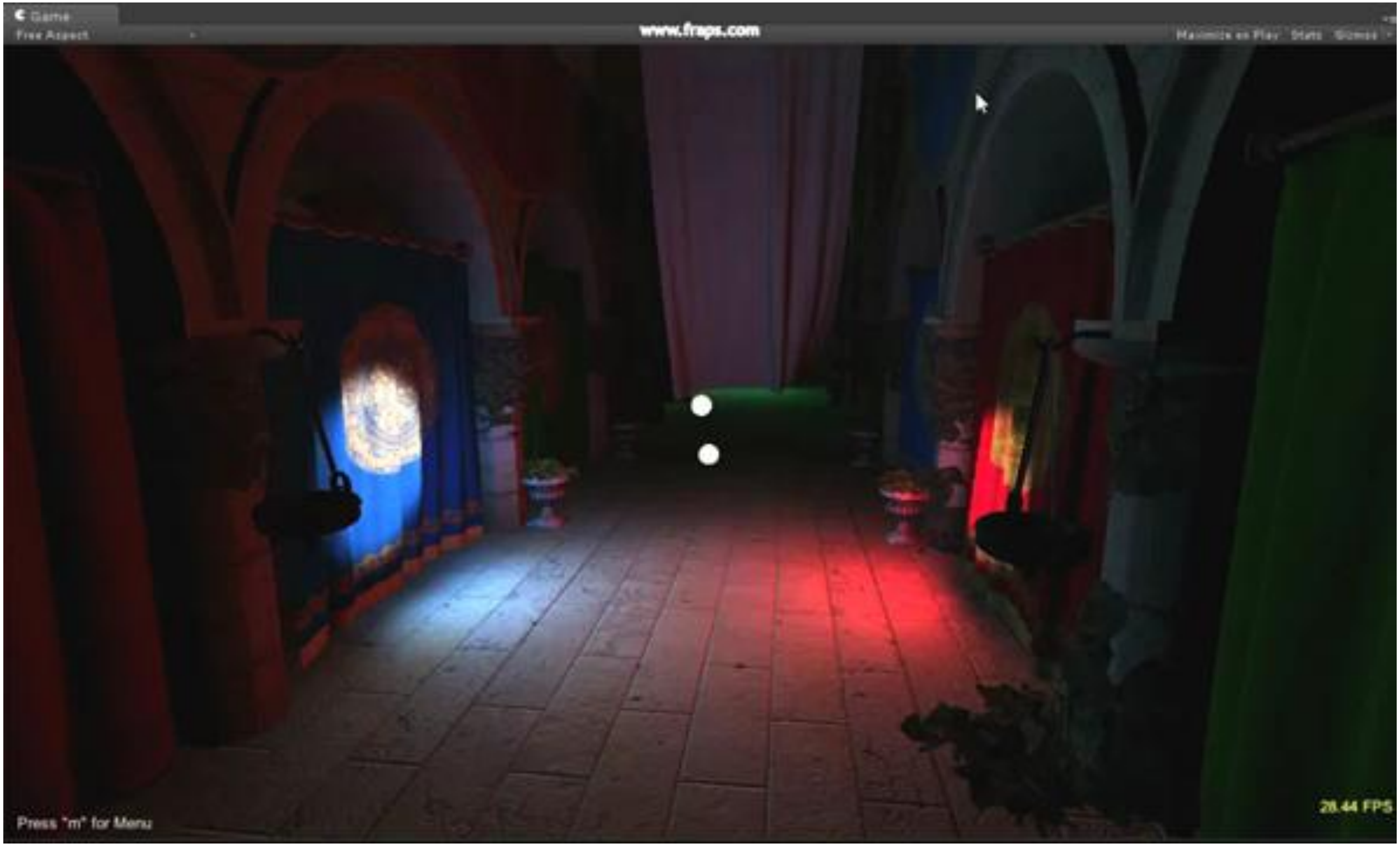
Reflective Shadow Map (cont.)

- Every pixel in the RSM represents an indirect light source
- If the resolution of RSM is 256 by 256, we got 65536 indirect light sources
- We can not afford to compute lighting from all pixels: **sampling**





Reflective Shadow Map (cont.)



END OF PRESENTATION
ANY QUESTIONS?