Summed-Area Variance Shadow Maps

Andrew Lauritzen ⋆ Michael McCool

Computer Graphics Lab, David R. Cheriton School of Computer Science, University of Waterloo

Abstract

Standard shadow maps suffer heavily from aliasing in part because of poor texture filtering. Common linear filtering algorithms such as mipmapping and summed-area tables are inapplicable to typical shadow maps which require a non-linear depth comparison per sample.

Variance shadow maps provide a solution to this problem by representing the shadow map in a way that can be linearly filtered. Unfortunately, they suffer from light bleeding. Additionally, arbitrary per-pixel filter sizes, which are desirable for plausible soft shadows algorithms, are not possible when pre-filtering a variance shadow map.

We address the light bleeding artifacts by making a small modification to the shadow attenuation function, and the filtering limitations by using summed-area tables.

Variance Shadow Maps

By rendering depth and squared depth into a shadow texture, we recover the moments M_1 and M_2 of the depth distribution over the texture filter region, from which we can compute:

$$\mu = E(x) = M_1$$
$$\sigma^2 = E(x^2) - E(x)^2 = M_2 - M_1^2$$

Finally we apply Chebyshev's Inequality to approximate the probability that a surface at depth t is in shadow:

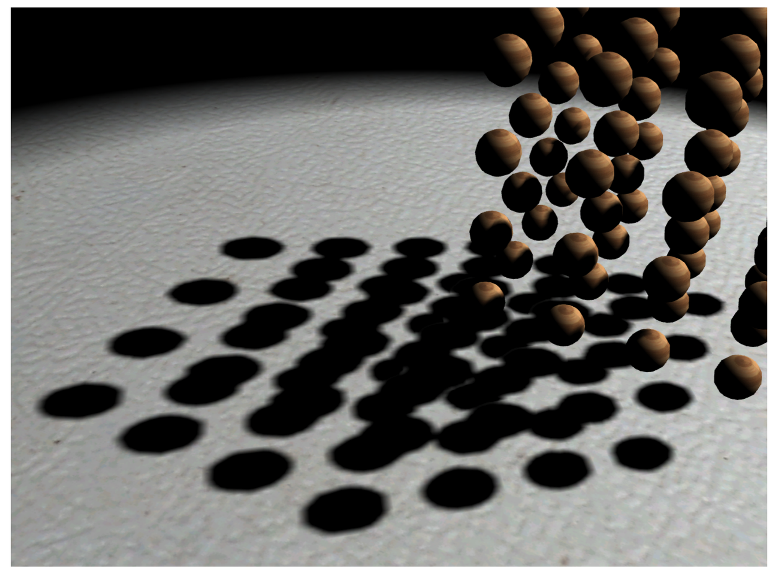
$$P(x \geq t) \approx p(t) = \frac{\sigma^2}{\sigma^2 + (t-\mu)^2}$$

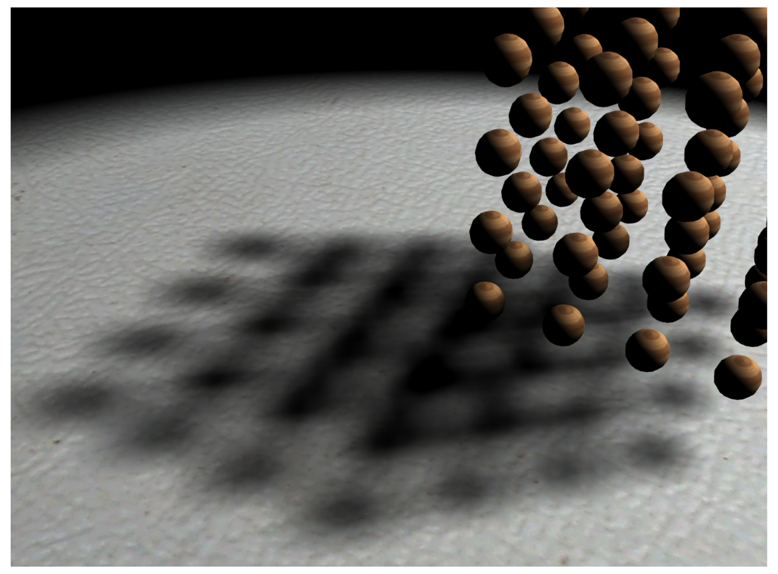
Blurring the variance shadow map before shading has the effect of clamping the minimum filter size, and produces uniform soft shadow edges.

Summed-Area Tables

Before shading, we generate a summed-area table from the shadow texture, which allows constant-time filtering of arbitrary rectangular regions. By using GPU derivative instructions we compute the filter size based on the pixel extents in texture space. We therefore achieve:

- High quality hardware-independent shadow filtering.
- Per-pixel filter size selection at constant cost.





Light Bleeding Reduction

The most objectionable light bleeding occurs in regions that should be fully occluded. We can reduce or eliminate these artifacts with a simple modification to $p(t)$, at the cost of over-darkening some legitimate penumbrae:

$$p'(t) = \max\left\{0, \frac{p(t)-\alpha}{1-\alpha}\right\}$$

where $0 \leq \alpha < 1$ is an aggressiveness parameter.

Performance

Even with fully dynamic updates, the performance of summed-area variance shadow maps is very good. This is particularly evident at larger filter sizes, for which percentage-closer filtering (PCF) is extremely slow. On a GeForce 8800 GTX at 1600×1200 with 4× MSAA we achieve the following frame times (lower is better):