Eurographics Symposium on Rendering 2013

20.6.2013

LINE-SWEEP AMBIENT OBSCURANCE

Ville Timonen

Åbo Akademi University

CONTENTS

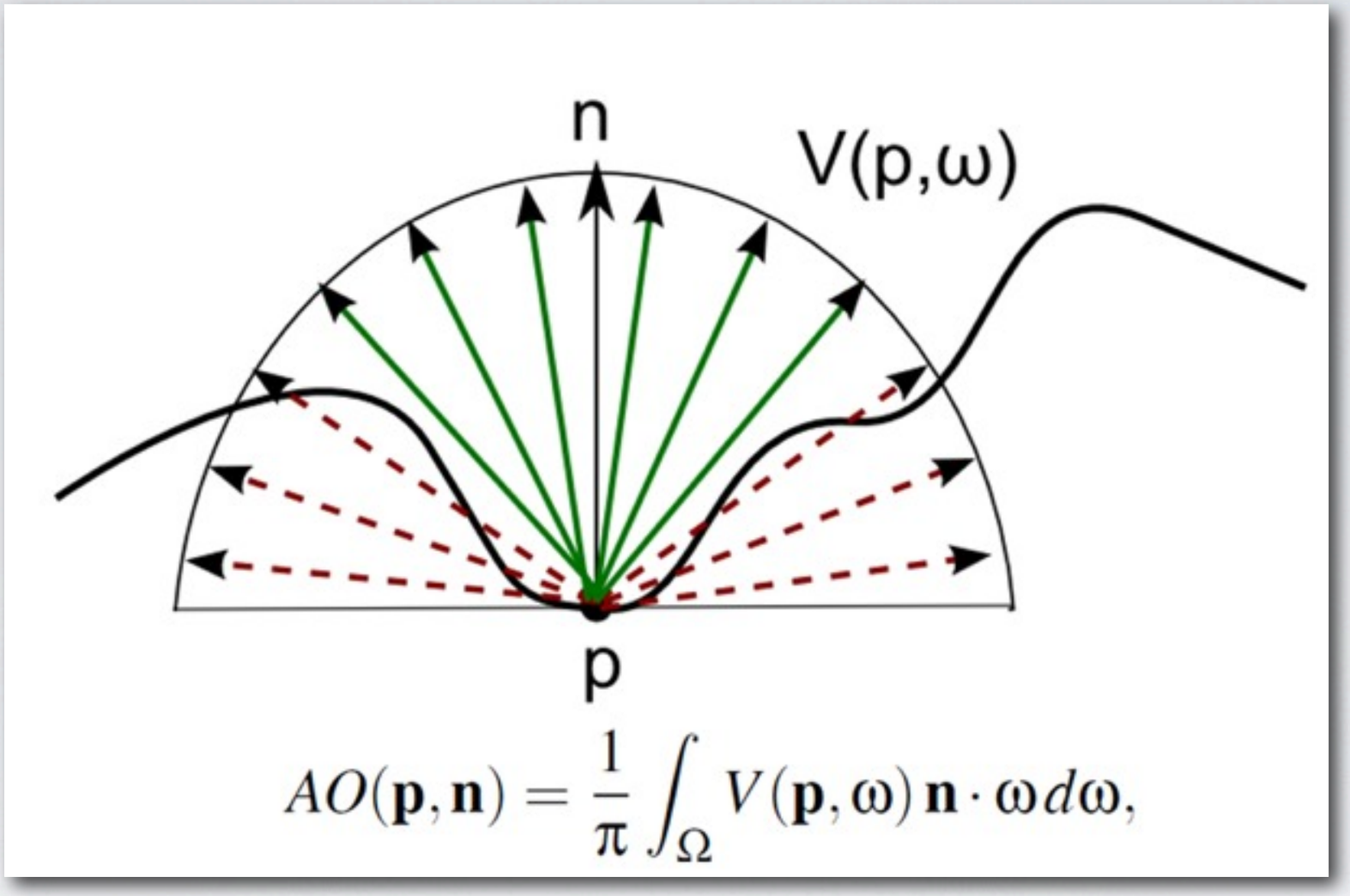
1. SSAO and previous approaches
2. Our method
3. Results
4. Questions

I AMBIENT OCCLUSION

Is an approximation for global illumination

I AMBIENT OCCLUSION

Defined as the cosine-weighted hemisphere visibility



$$AO(\mathbf{p},\mathbf{n}) = \frac{1}{\pi}\int_{\Omega} V(\mathbf{p},\omega)\,\mathbf{n}\cdot\omega\,d\omega,$$

V is a binary visibility function
(1 = hits geometry, 0 = does not hit geometry)

1 AMBIENT OBSCURANCE

Obscurance: Also account for the distance of the geometry

- Add a falloff term F that tapers off as a function of distance
- F(0) = 1, F(inf) = 0

$$A(\mathbf{p}, \vec{n}) = \frac{1}{\pi} \int_{\Omega} F(D(\mathbf{p}, \vec{\omega}))\, \vec{n} \cdot \vec{\omega}\, d\vec{\omega}$$

- Now need to know distance to occluder, D, instead

1 AMBIENT OBSCURANCE

Ideal solution

- Ideally solved at each fragment, against all scene geometry
- Problem 1: Dependent on scene complexity and scenes can get prohibitively large
- Problem 2: Generic geometry does not allow efficient acceleration
- Think about a path tracer that shoots hundreds of rays at each pixel: Too slow for real-time today

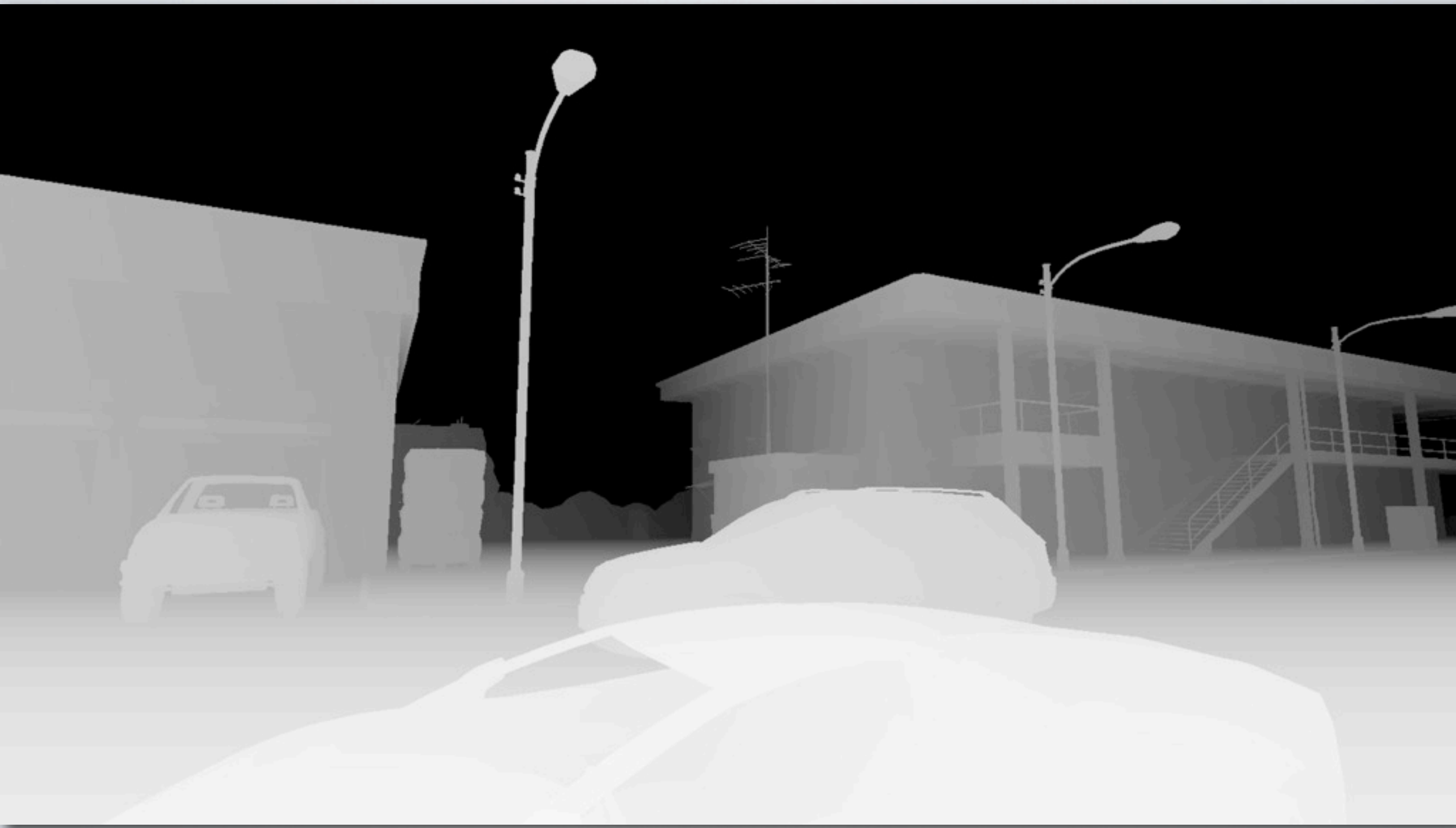
1 AMBIENT OBSCURANCE

- What can we afford, then?
- ~2000 clock cycles per pixel per frame, or ~60 bilinear texture fetches *at most* (assuming 1080p, 5ms/frame)

→ Screen-Space approximations of Ambient Obscurance

I SCREEN-SPACE AMBIENT OBSCURANCE

This is a depth map (dark = far, light = near)



1 SCREEN-SPACE AMBIENT OBSCURANCE

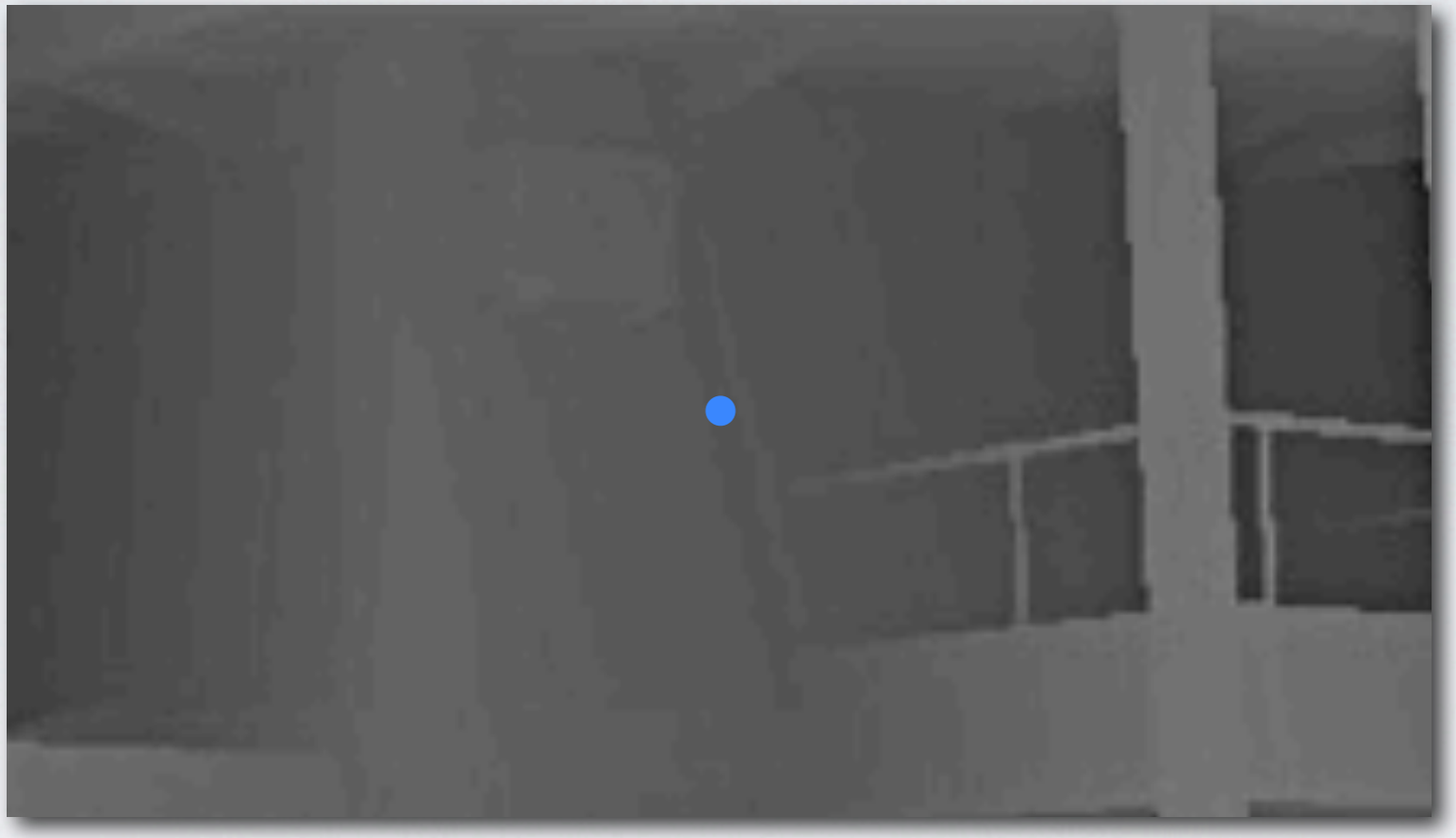
The depth map represents incomplete scene geometry...

- Because we don't know what's behind the first depth layer
- Because we don't know what's outside the view frustum
- Oh well, can't have it all... :-(

I SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:

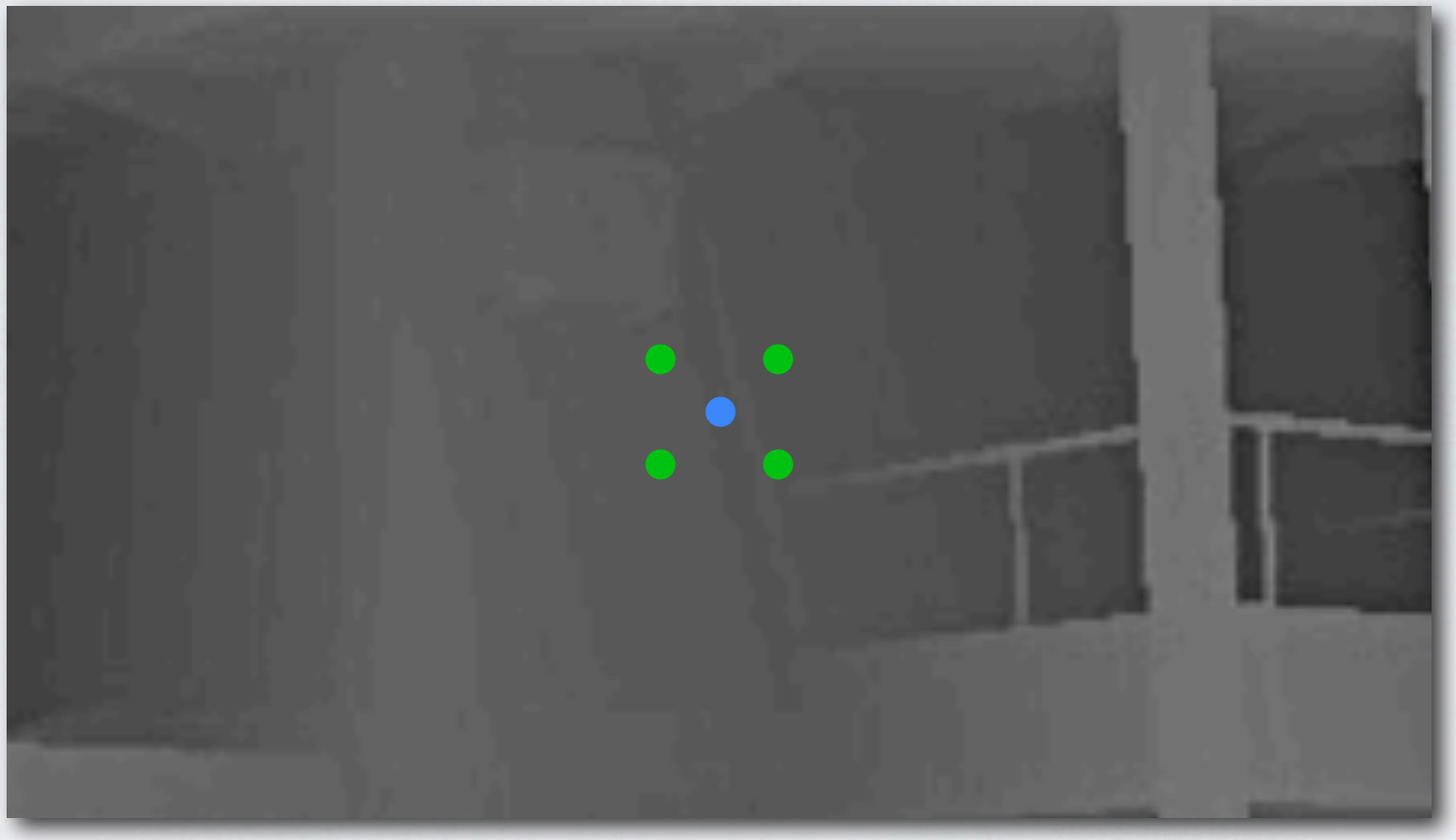
For the blue point...



I SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:

For the blue point...

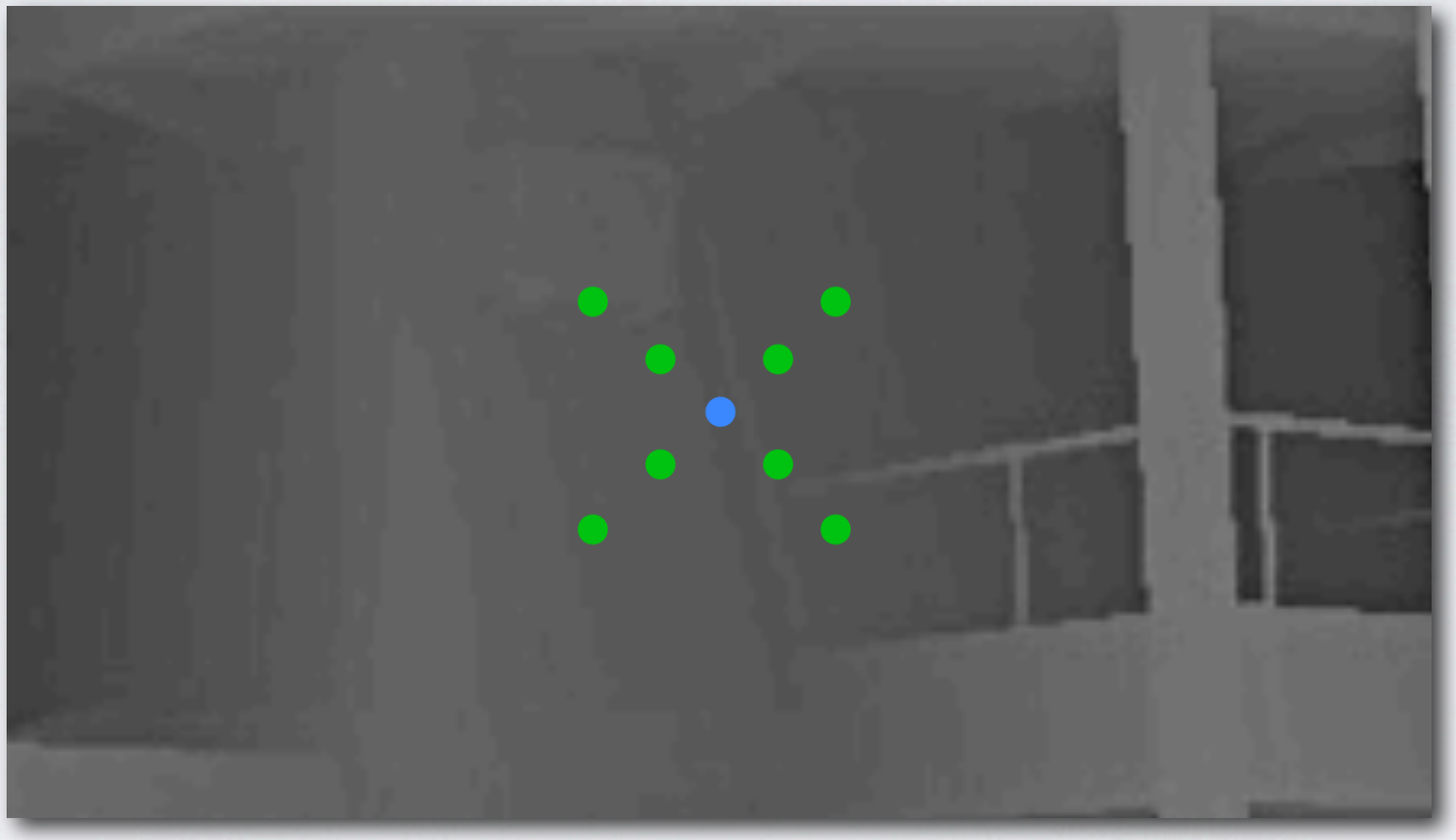


Sample the surroundings...

I SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:

For the blue point...

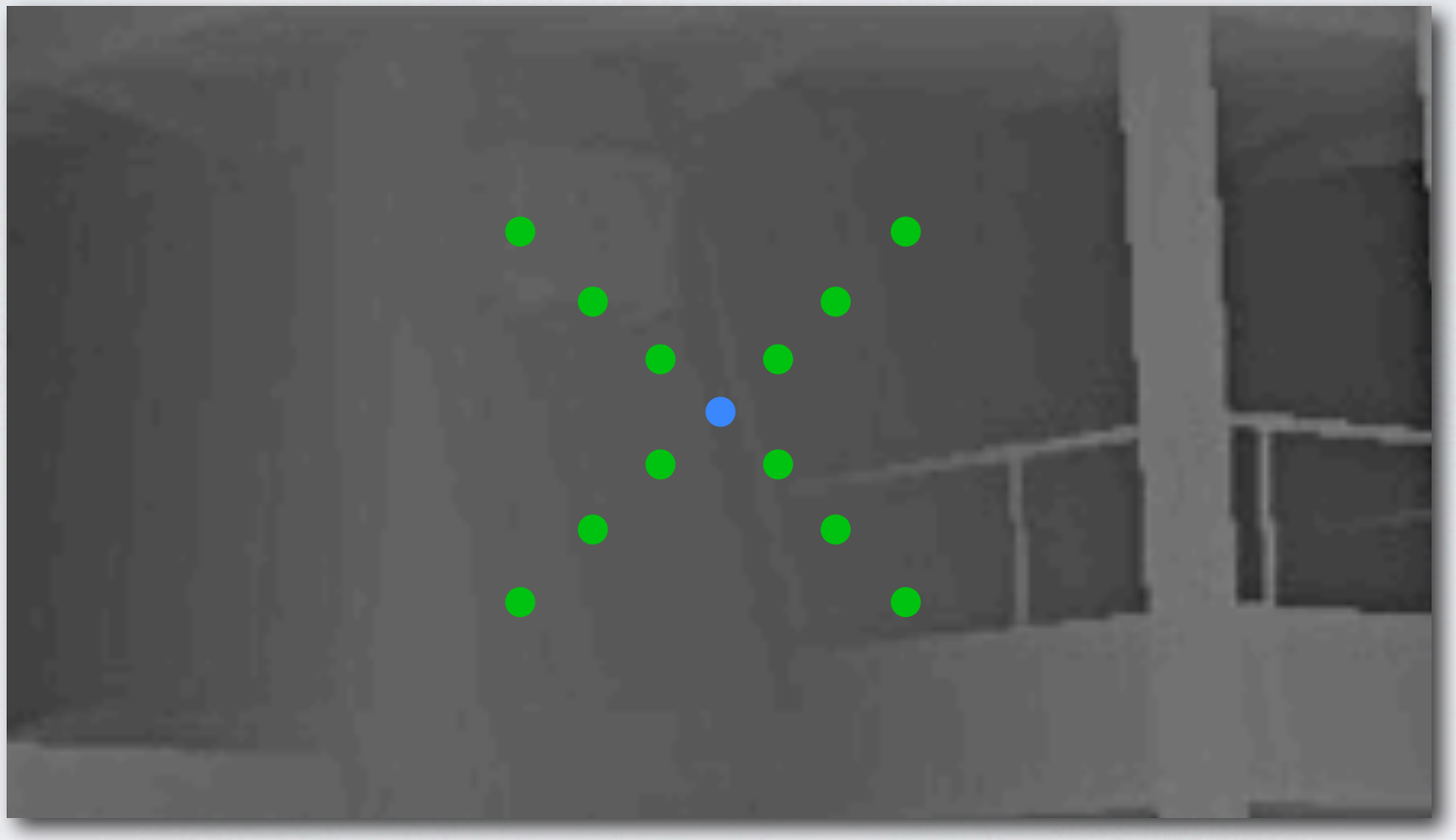


Sample the surroundings...

I SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:

For the blue point...

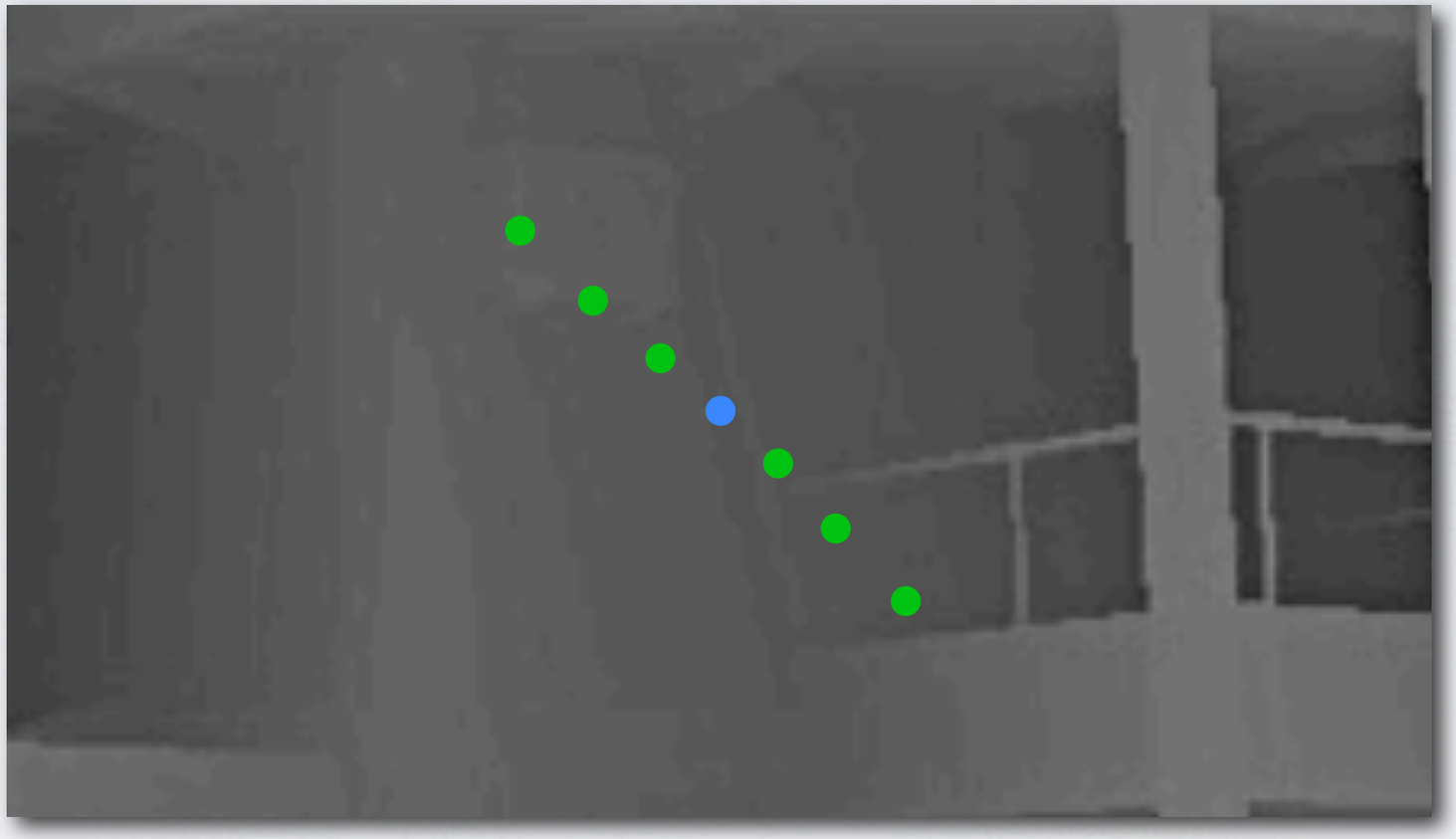


Sample the surroundings...

I SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:

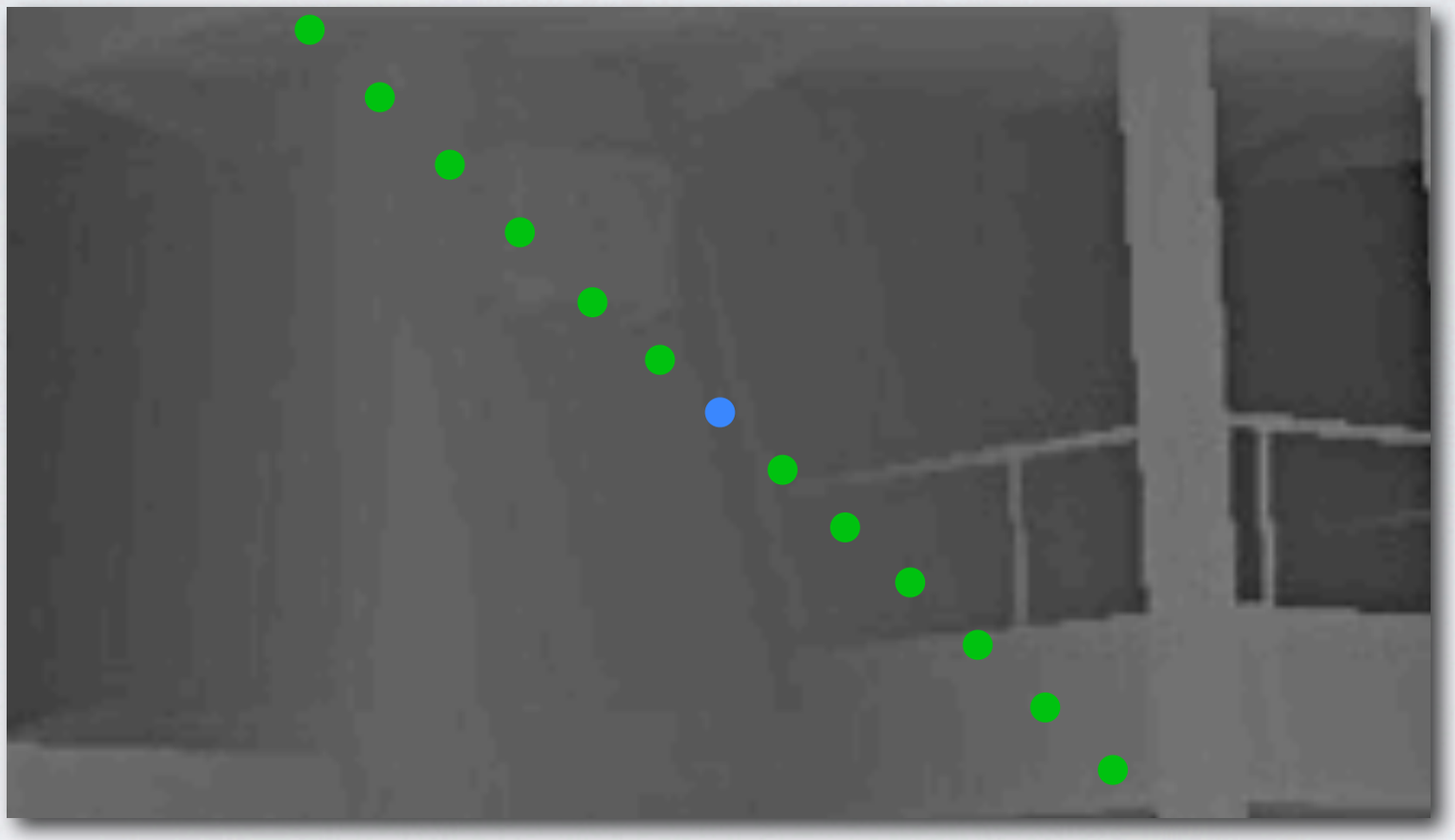
But pay attention to one sampling line..



I SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:

But pay attention to one sampling line..

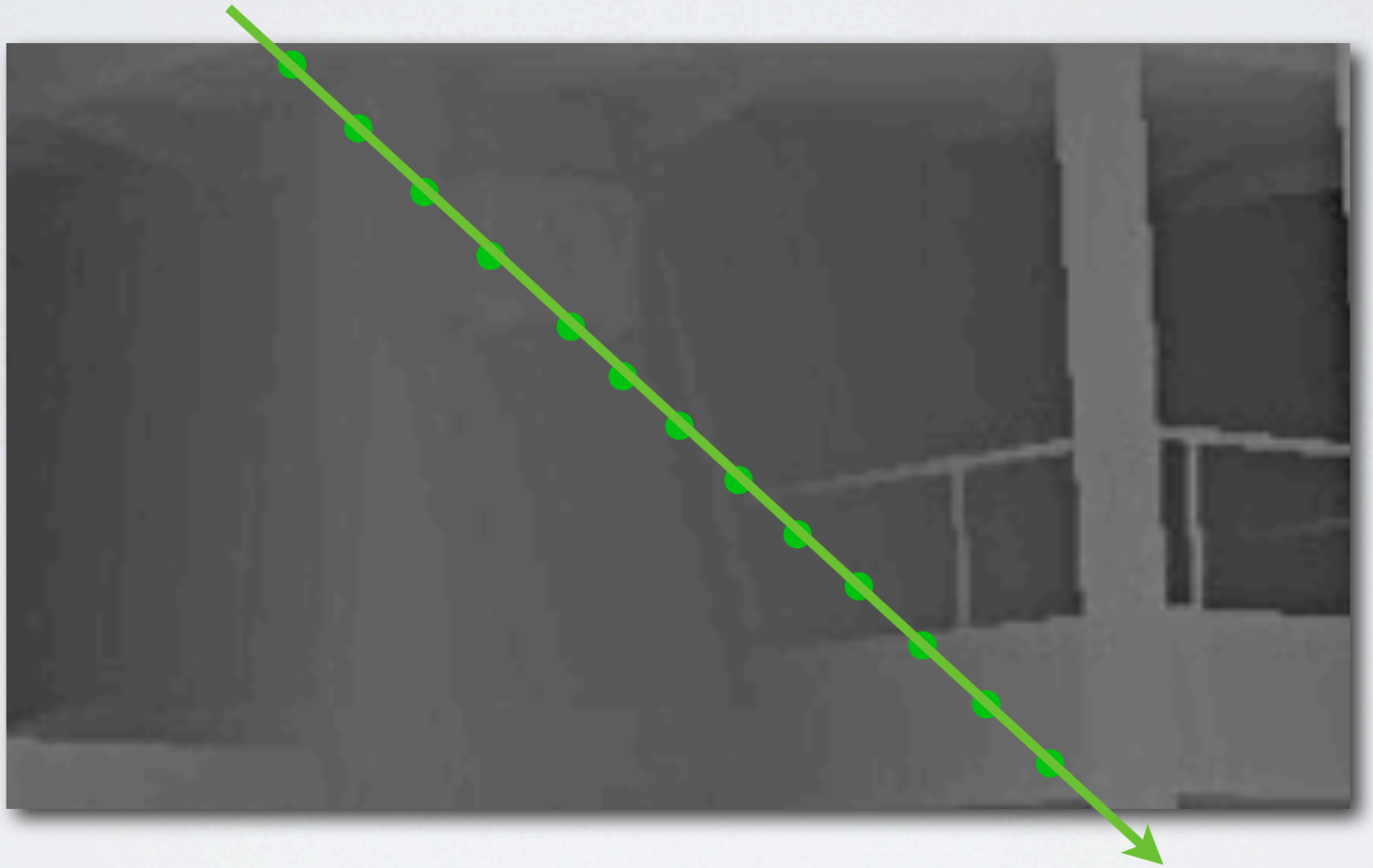


All receivers along the line go through the same data!

2 OUR METHOD

Motivation

This gives rise to an idea:

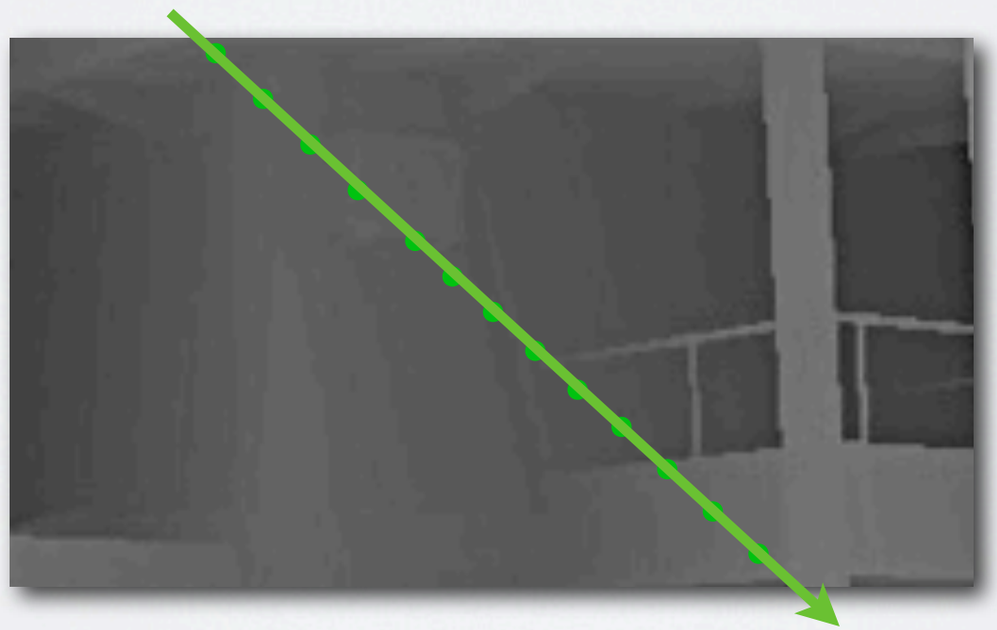


Go through these points in one line sweep

2 OUR METHOD

The main idea

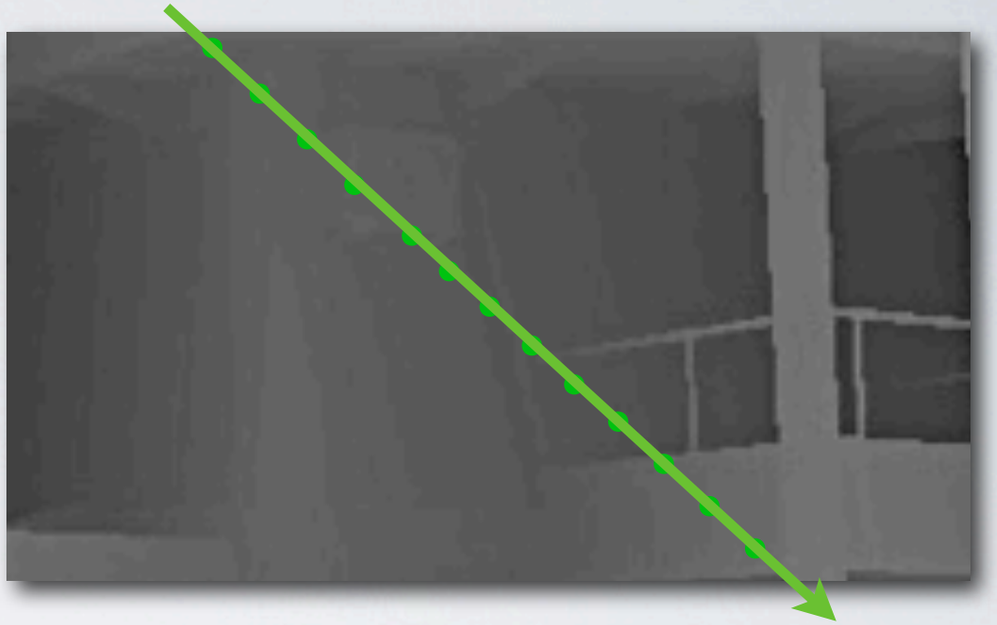
- We sweep through the line one step at a time
- We maintain an internal representation of the geometry along the line so far
- From this representation, we extract AO for each successive point



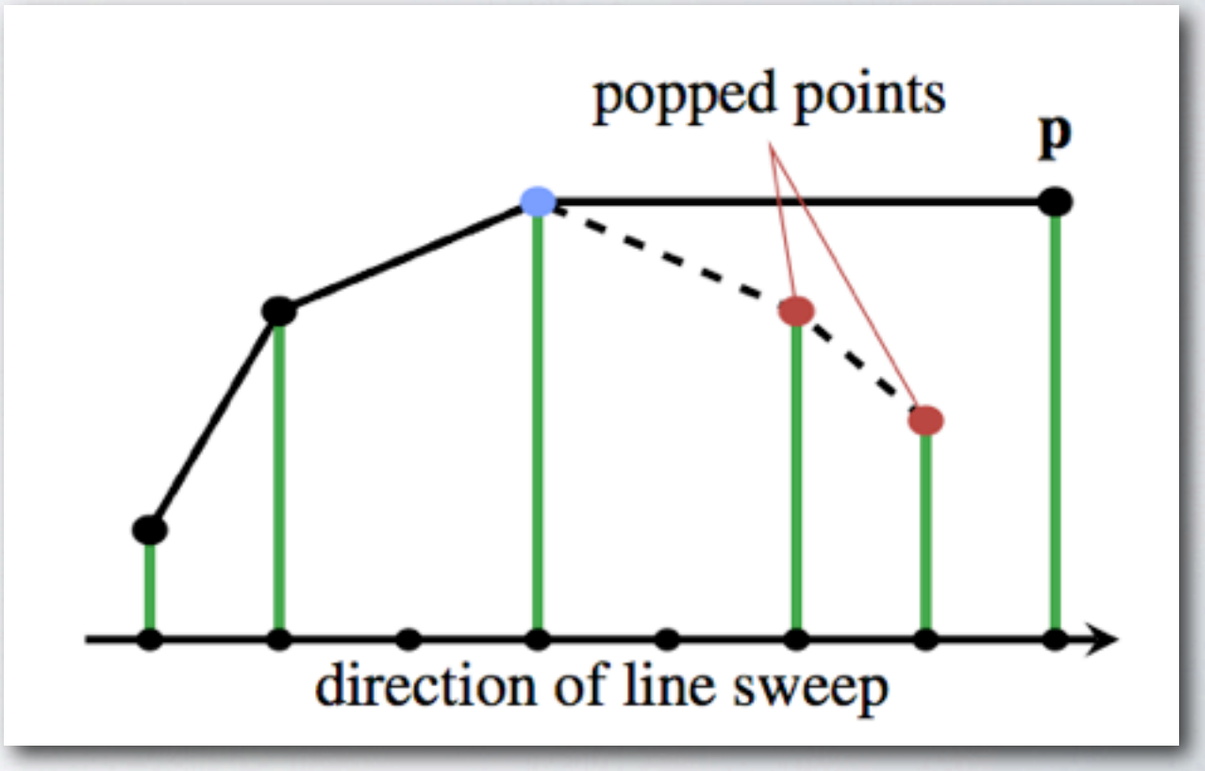
2 OUR METHOD

The internal data structure

- The internal data structure is a stack
- Holds a subset of the visited points



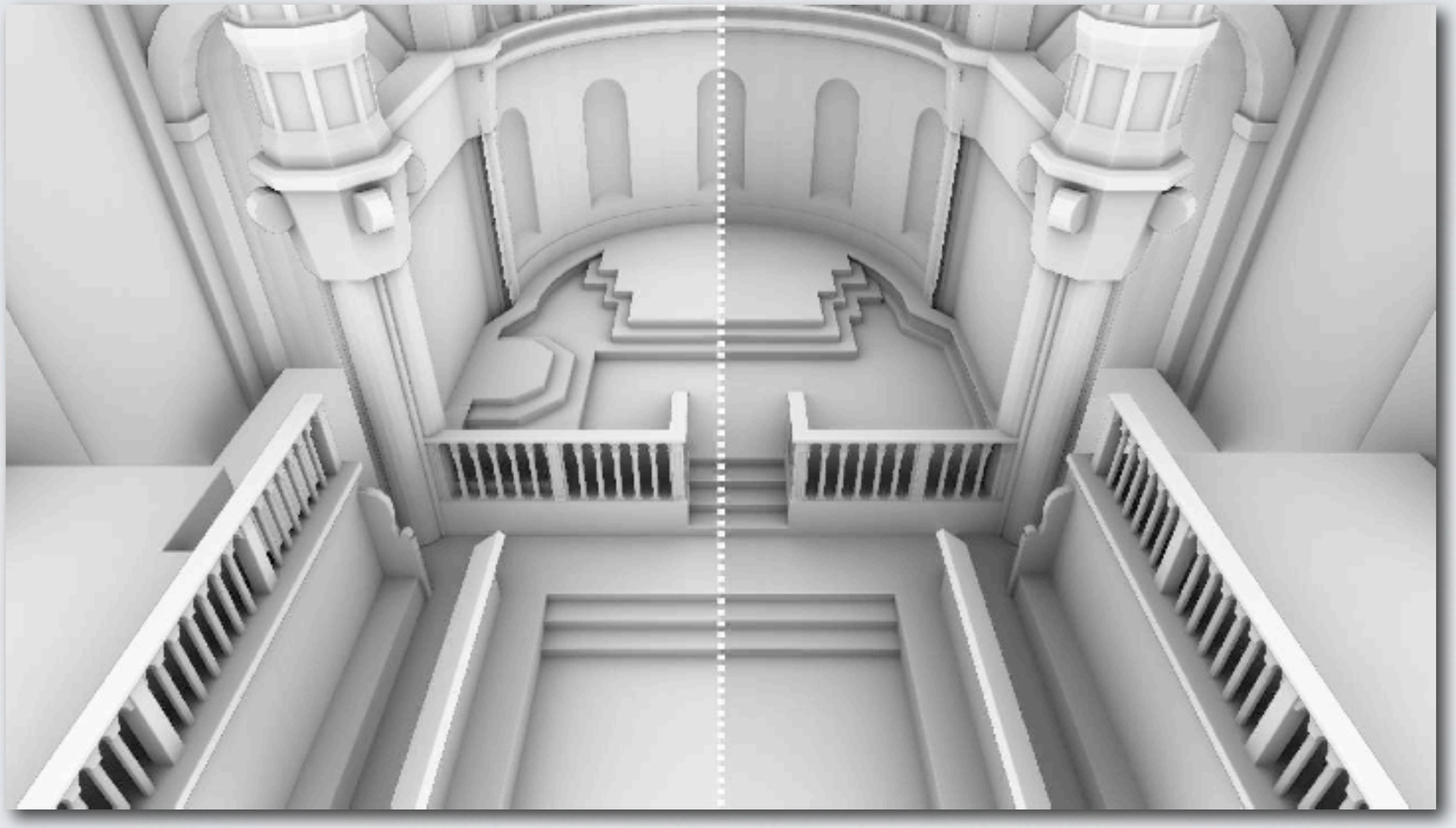
- Pop until the largest falloff-weighted occluder is at the top
- Evaluate AO from that occluder
- Push in the new point **p** and continue



2 OUR METHOD

Largest occluder

We only consider the largest occluder, but this is acceptable:

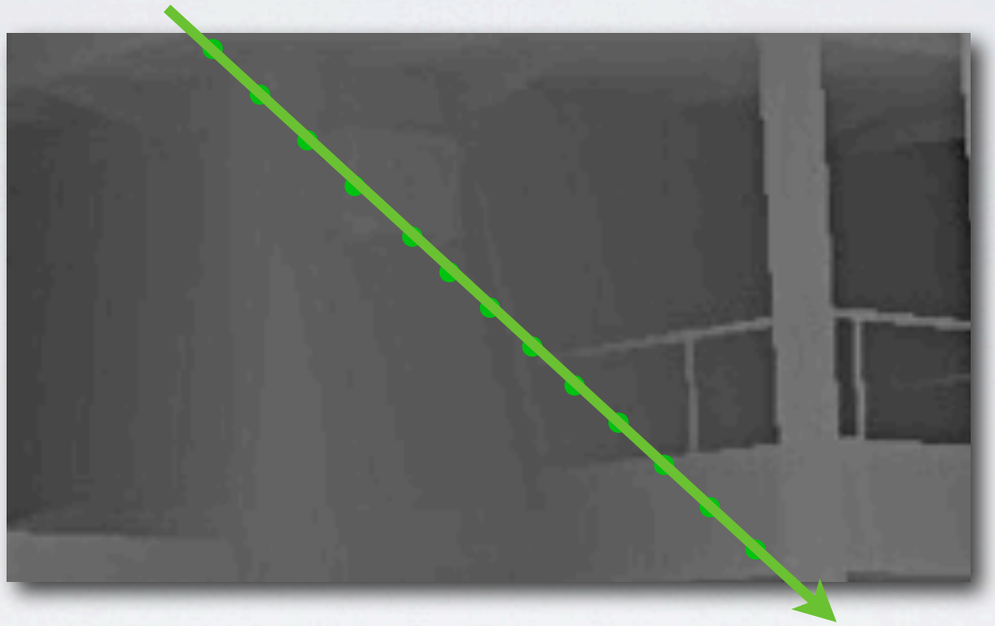


Largest occluder All occluders

2 OUR METHOD

Time complexity

- Processing a line of N samples is O(N)



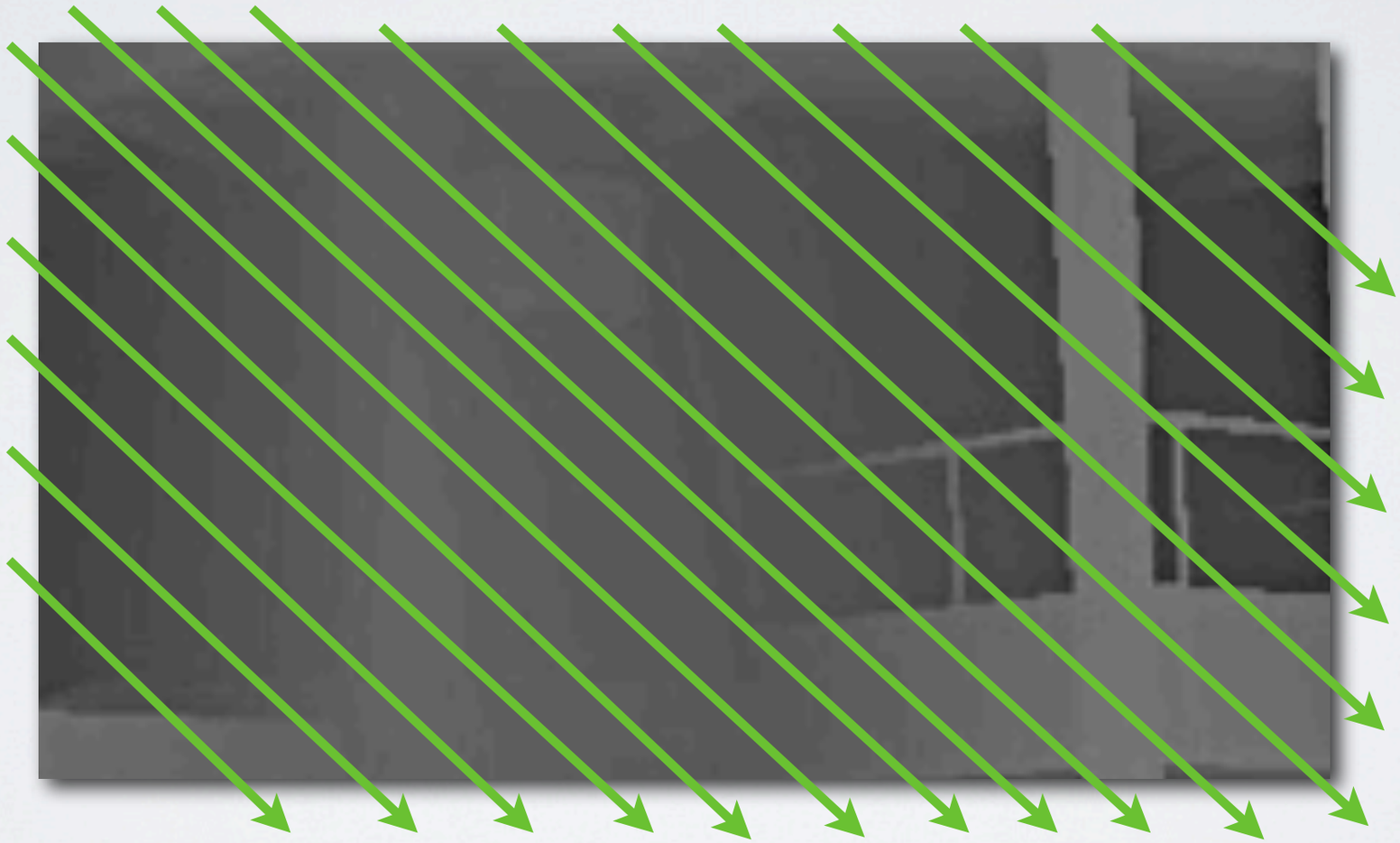
- Amortized constant cost O(1) per pixel per screen direction

- Previous methods, e.g. HBAO, takes multiple (say, 32) iterations for HQ

2 OUR METHOD

Reaching the final result

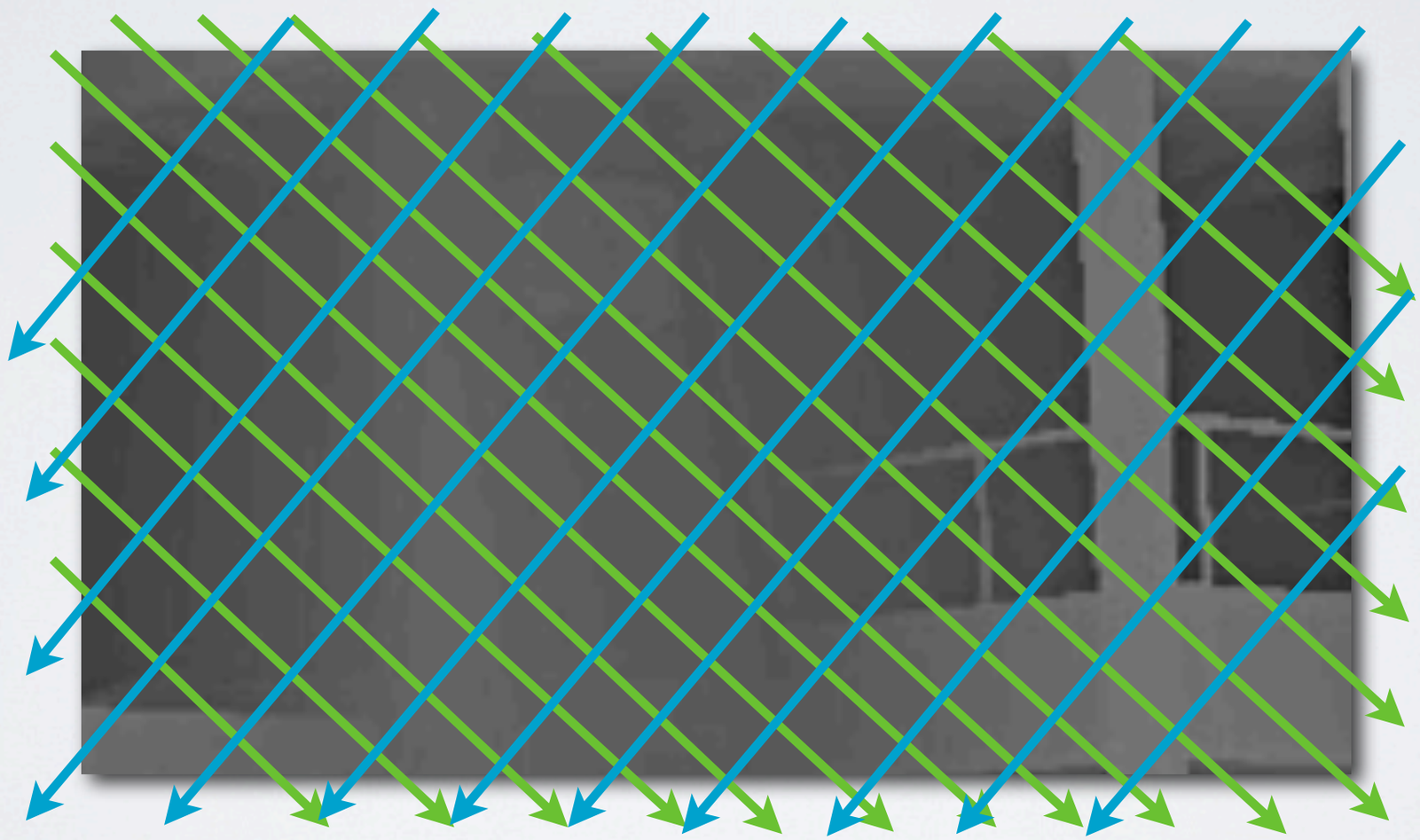
- Cover the whole framebuffer with lines...



2 OUR METHOD

Reaching the final result

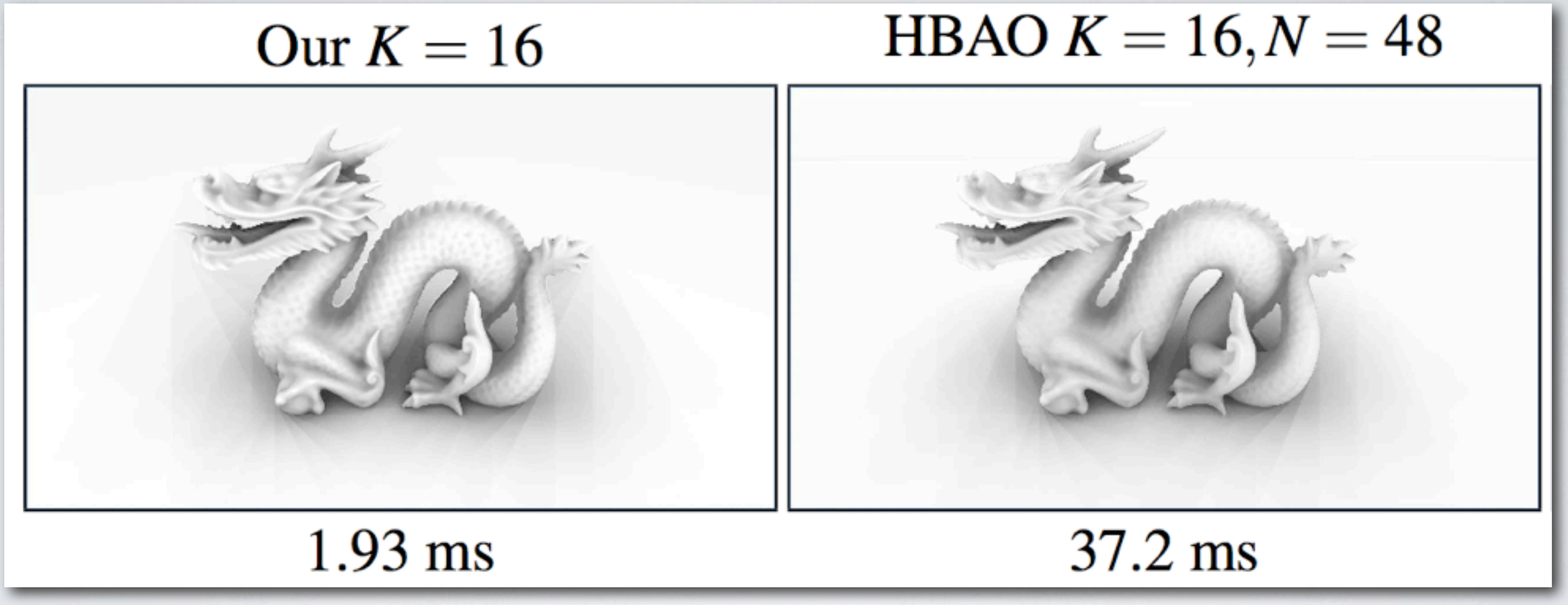
- Cover the whole framebuffer with lines...



- ...in multiple directions, and gather the results per screen pixel

3 RESULTS

1280(+256)×720(+144), GTX480



3 RESULTS

1280(+256)×720(+144), GTX480

Our $K = 16$

2.56 ms

HBAO $K = 16, N = 32$

24.2 ms

3 RESULTS

Scaling

Screen resolution	Our method	HBAO
800×600	1.49 ms	10.5 ms
1280×720	2.56 ms	24.2 ms
1920×1080	5.24 ms	92.5 ms
2560×1600	9.58 ms	249 ms

3 RESULTS

1920(+384)x1080(+216), 4.9ms/frame, GTX480

Line-Sweep Ambient Occlusion

4 QUESTIONS

Or comments...