Scalable SSAO algorithms

Ville Timonen

SyysGraph 2013

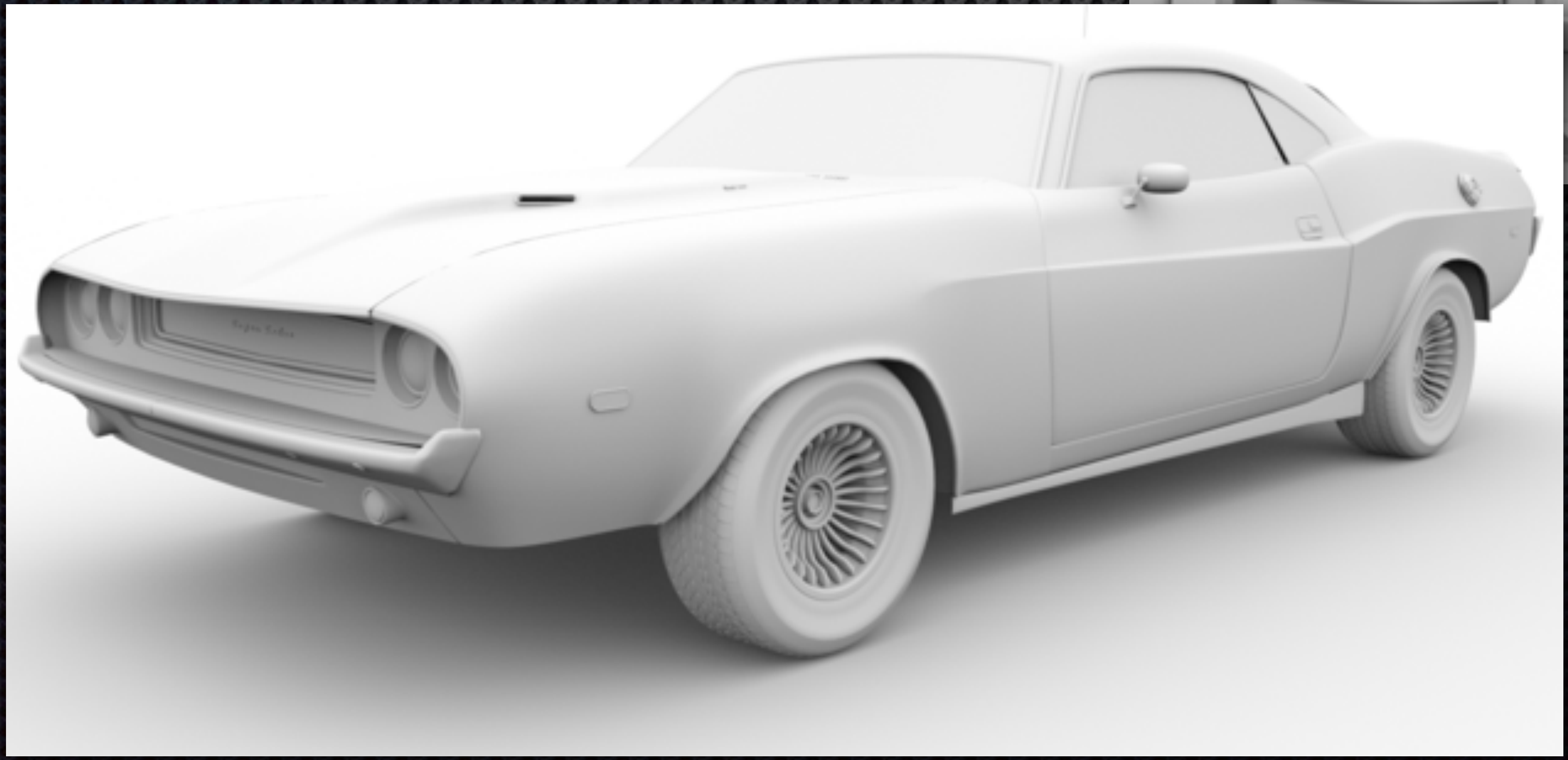
CONTENTS

1. SSAO and previous approaches
2. Line-Sweep Ambient Obscurance (EGSR’13)
3. Far-Field Screen-Space Ambient Obscurance (HPG’13)
4. Questions

1 AMBIENT OBSCURANCE

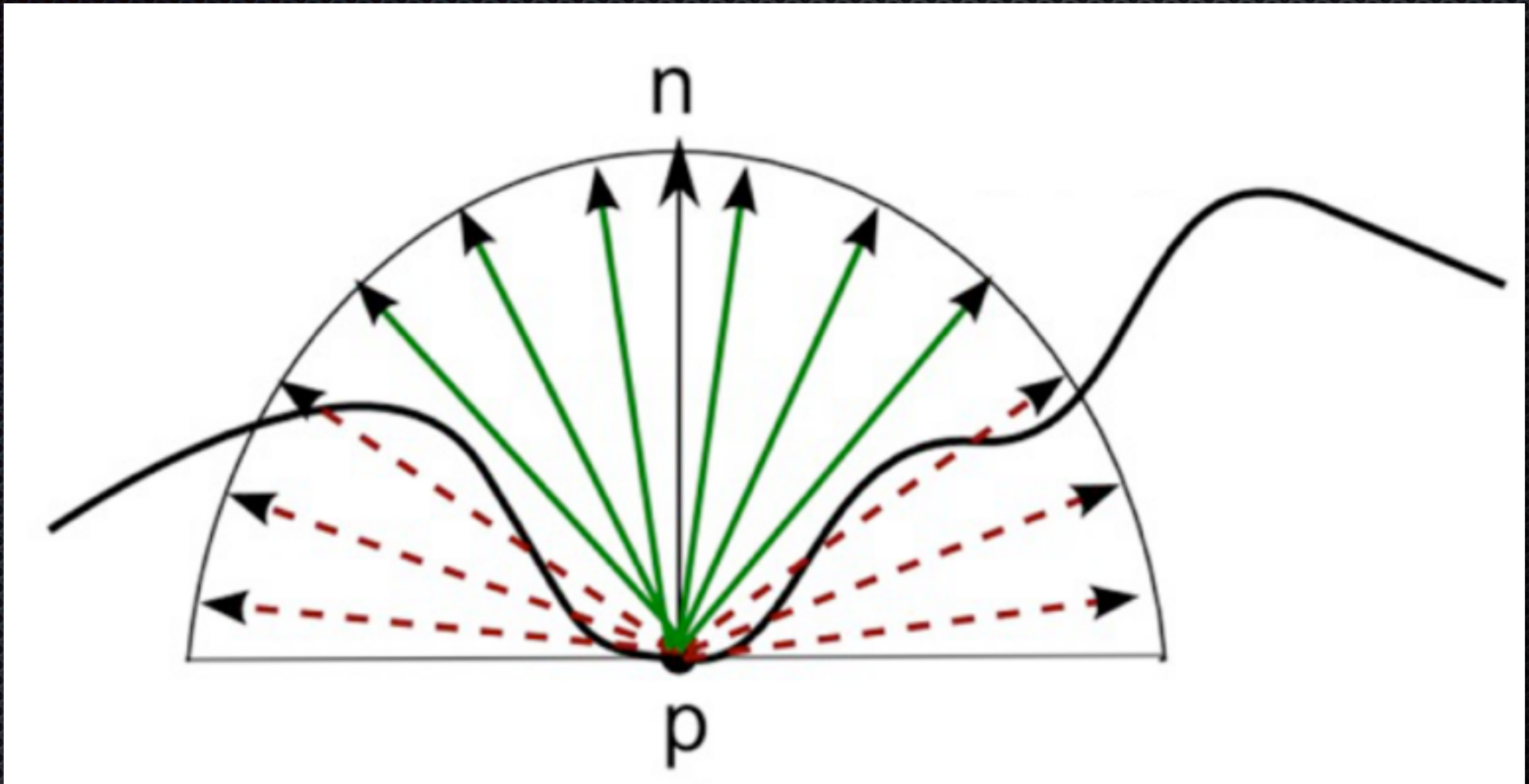
Is an approximation for global illumination

AdamLHumphreys.com



1 AMBIENT OBSCURANCE

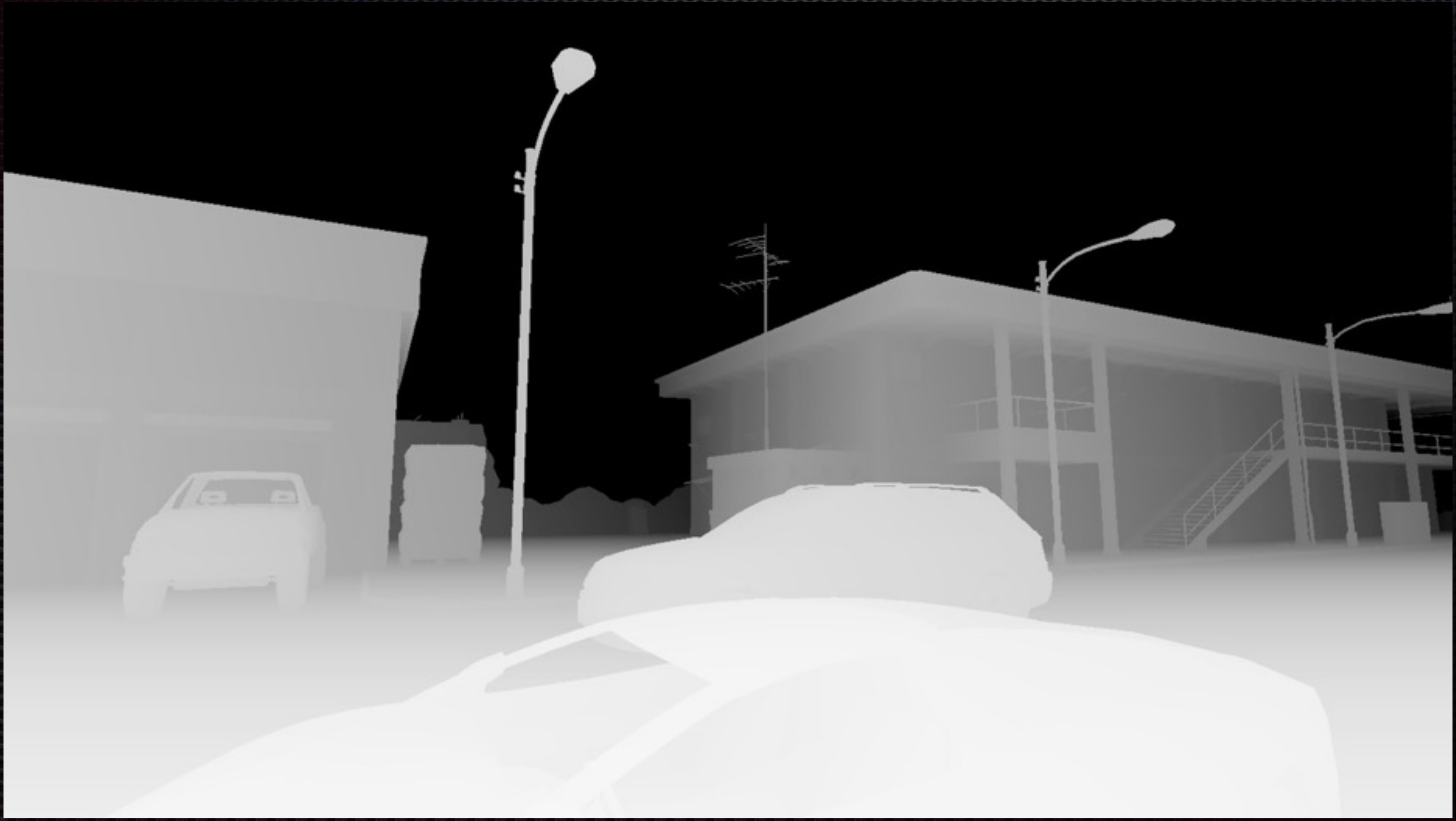
Defined as the cosine and falloff weighted hemisphere visibility



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

I SCREEN-SPACE AMBIENT OBSCURANCE

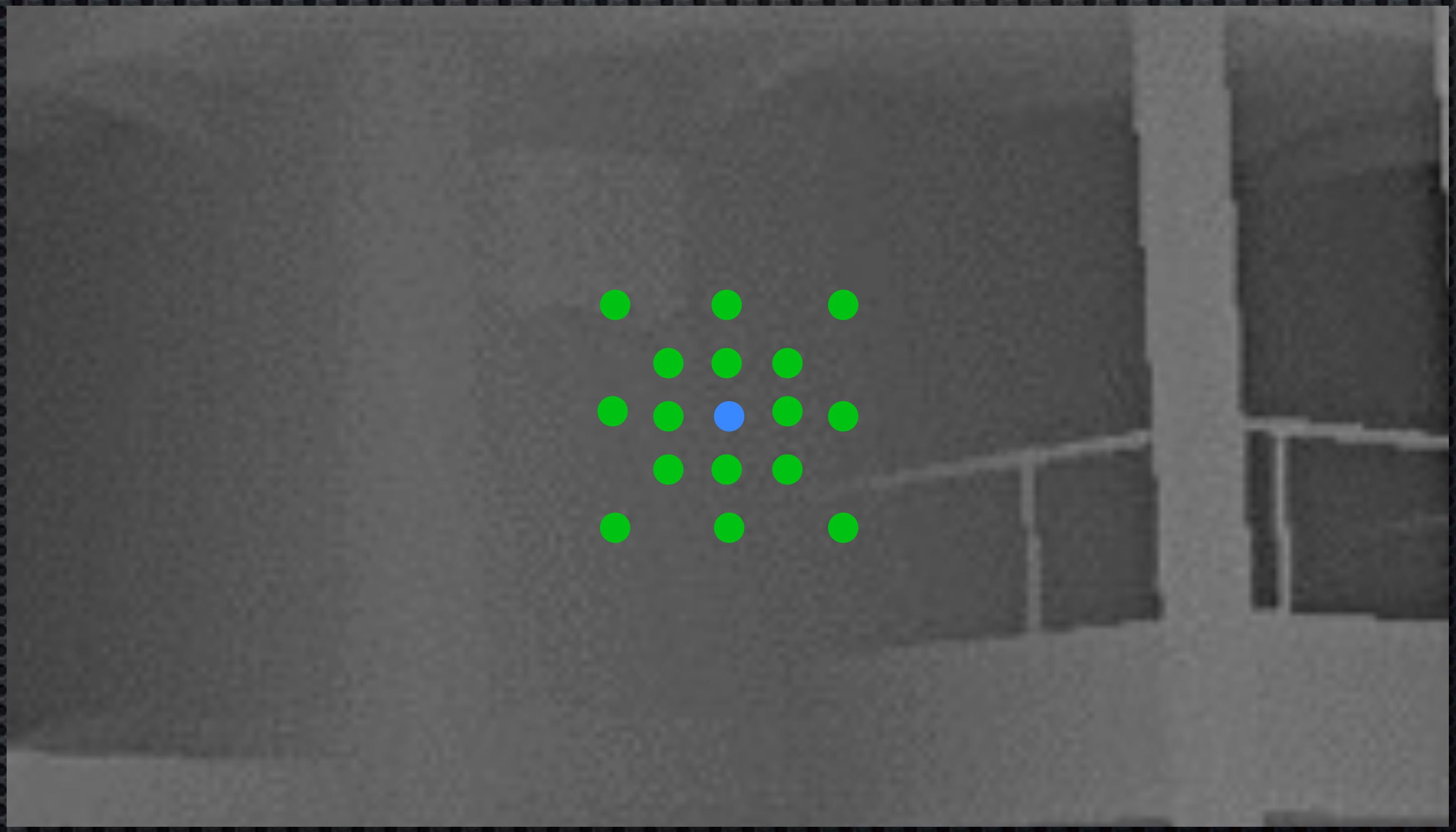
This is a depth map (dark = far, light = near)



A by-process of most graphics pipelines

So this is what SSAO does:

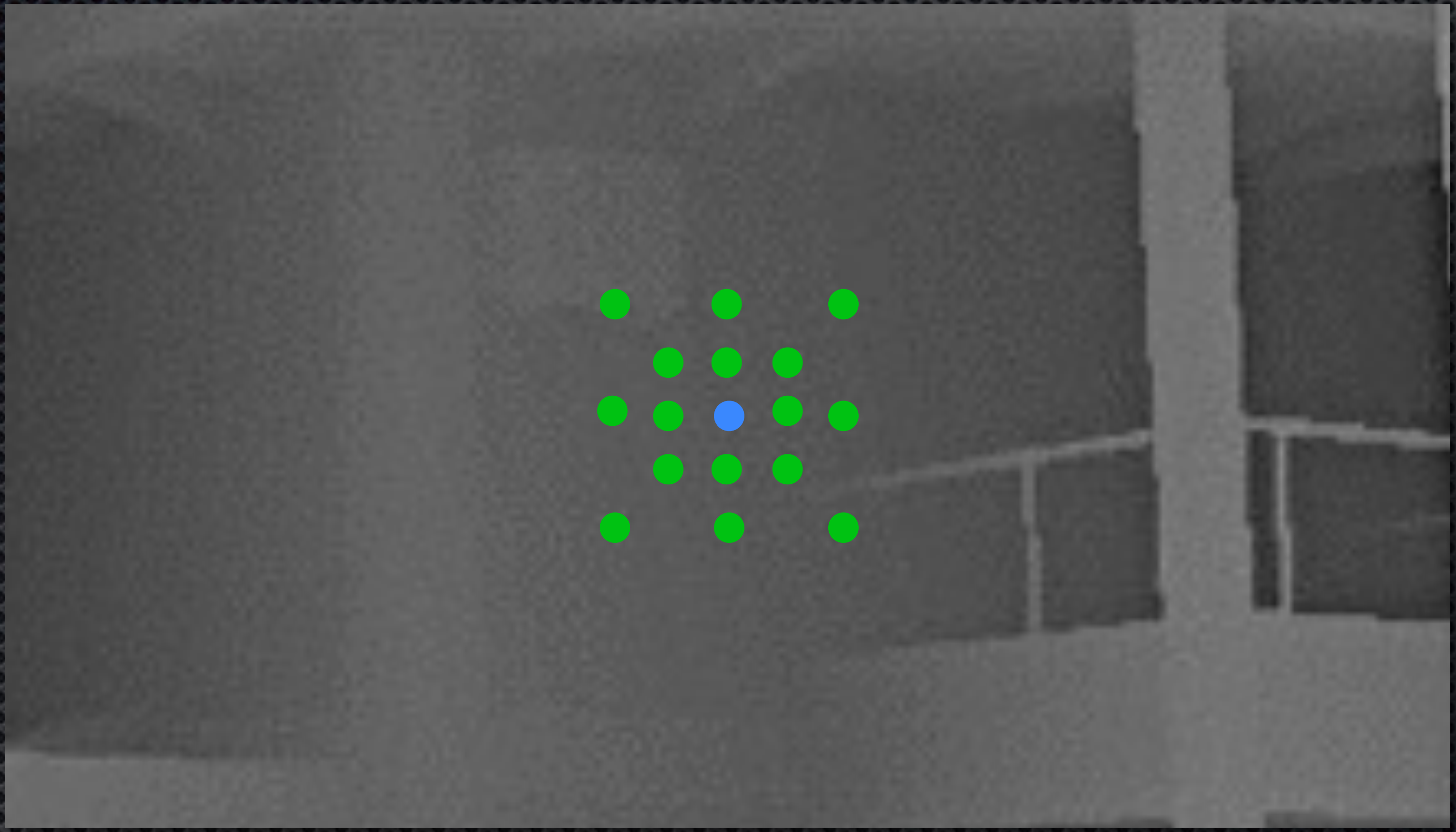
For the blue point...



Sample the surroundings...

1 SCREEN-SPACE AMBIENT OBSCURANCE

So this is what SSAO does:



Deproject points into world space and evaluate AO:

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

I SCREEN-SPACE AMBIENT OBSCURANCE

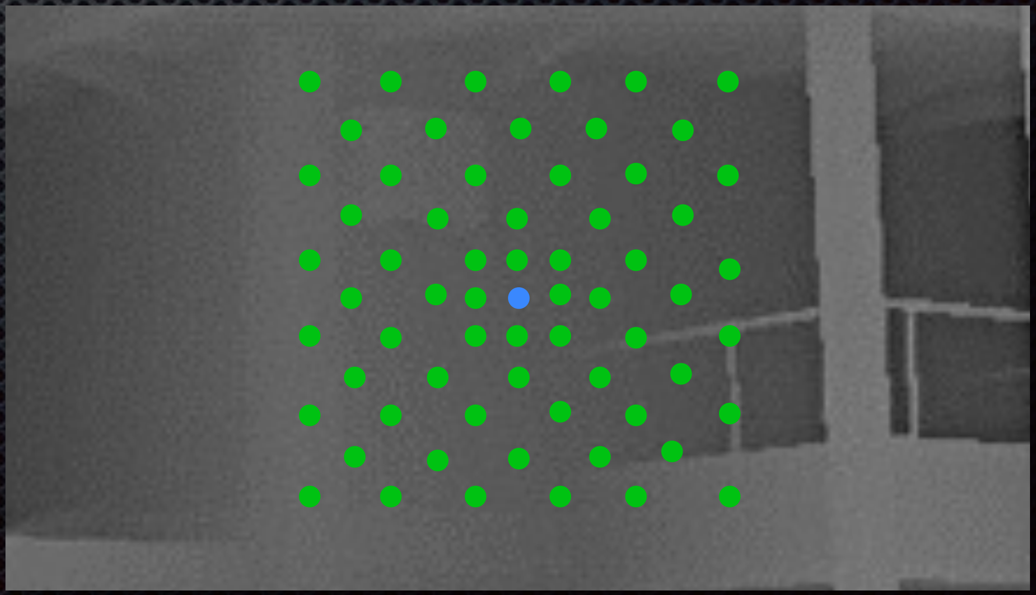
Traditional SSAO issues:

The amount of sampling limits scalability

Can be battled by:

- Sparse sampling, which produces noise
- Noise can be traded for blur, which corrupts details

In any case, has trouble scaling beyond very local effects



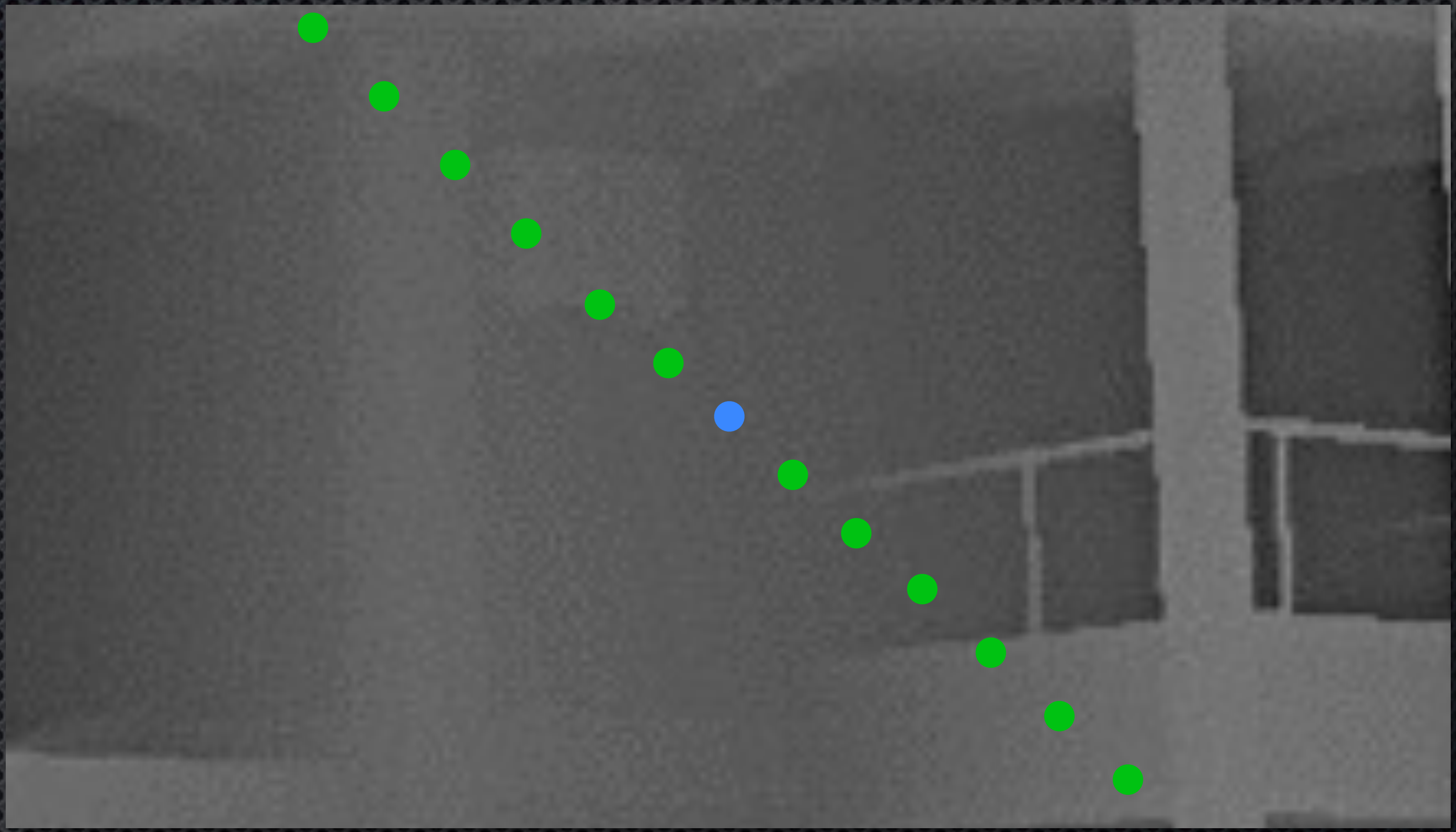
LINE SWEEP AMBIENT OBSCURANCE

Eurographics Symposium on Rendering 2013
Computer Graphics Forum 32(4)
Best Student paper

Ville Timonen

2 LINE SWEEP AMBIENT OBSCURANCE

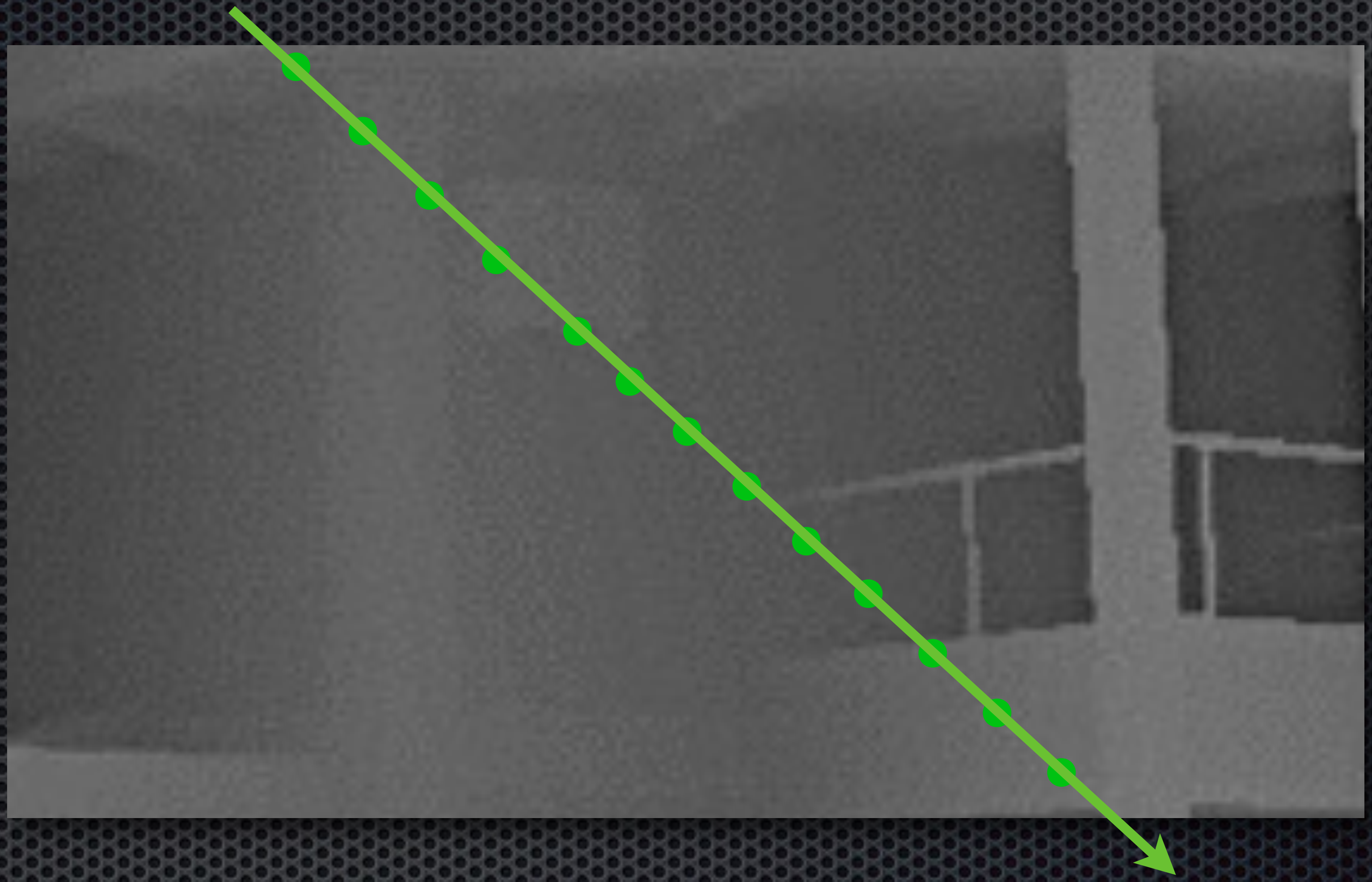
Consider samples along one line



All receivers along the line go through the same data!

2 LINE SWEEP AMBIENT OBSCURANCE

This gives rise to an idea:

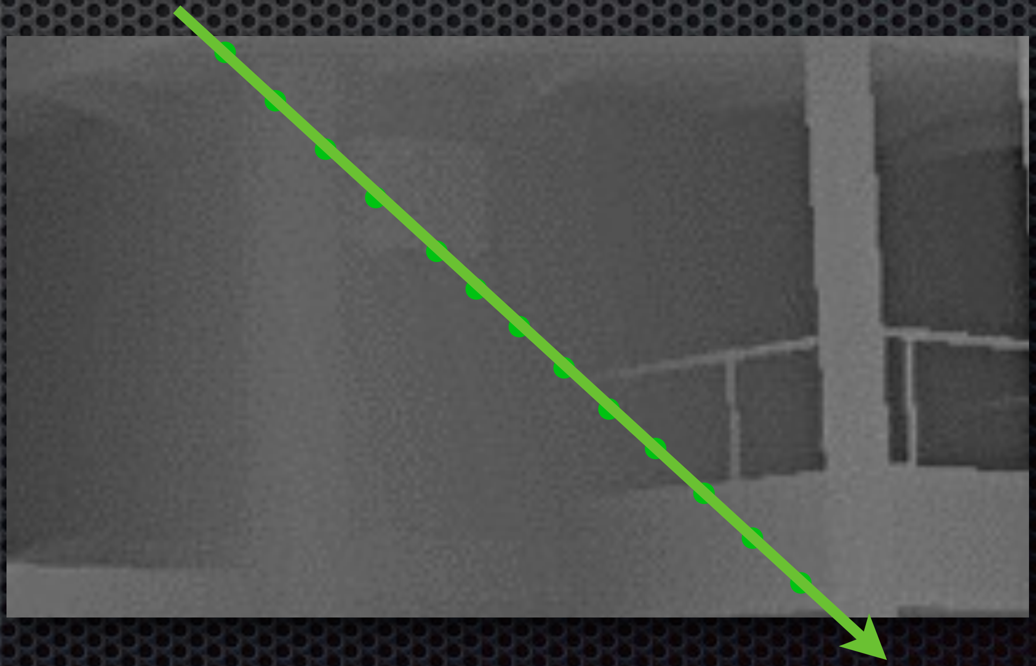


Go through these points in one line sweep

2 LINE SWEEP AMBIENT OBSCURANCE

The main idea

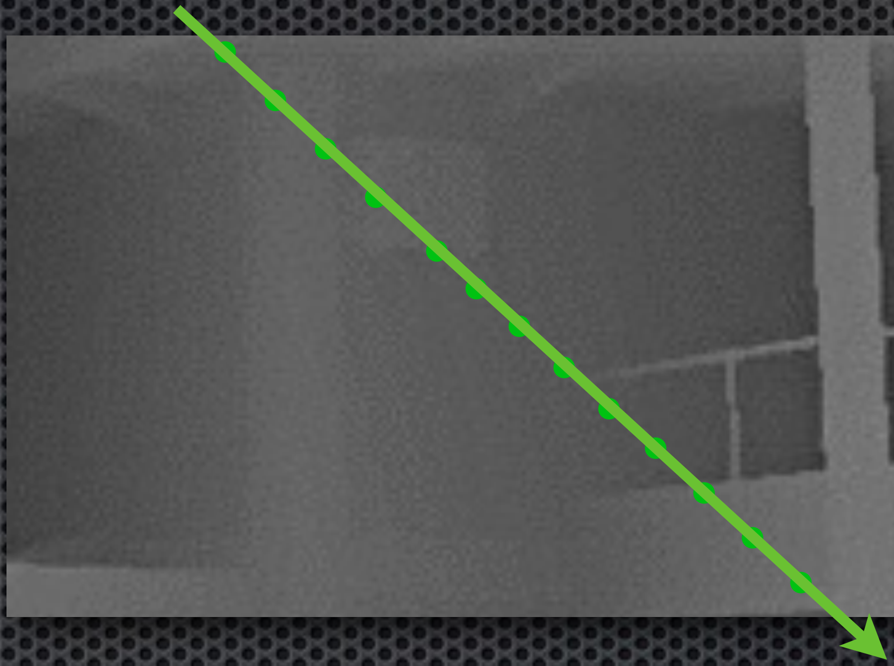
- We sweep through the line one step at a time
- We maintain an internal representation of the geometry along the line so far: A stack of the visited points
- From this representation, we extract AO for each successive point



2 LINE SWEEP AMBIENT OBSCURANCE

Time complexity

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

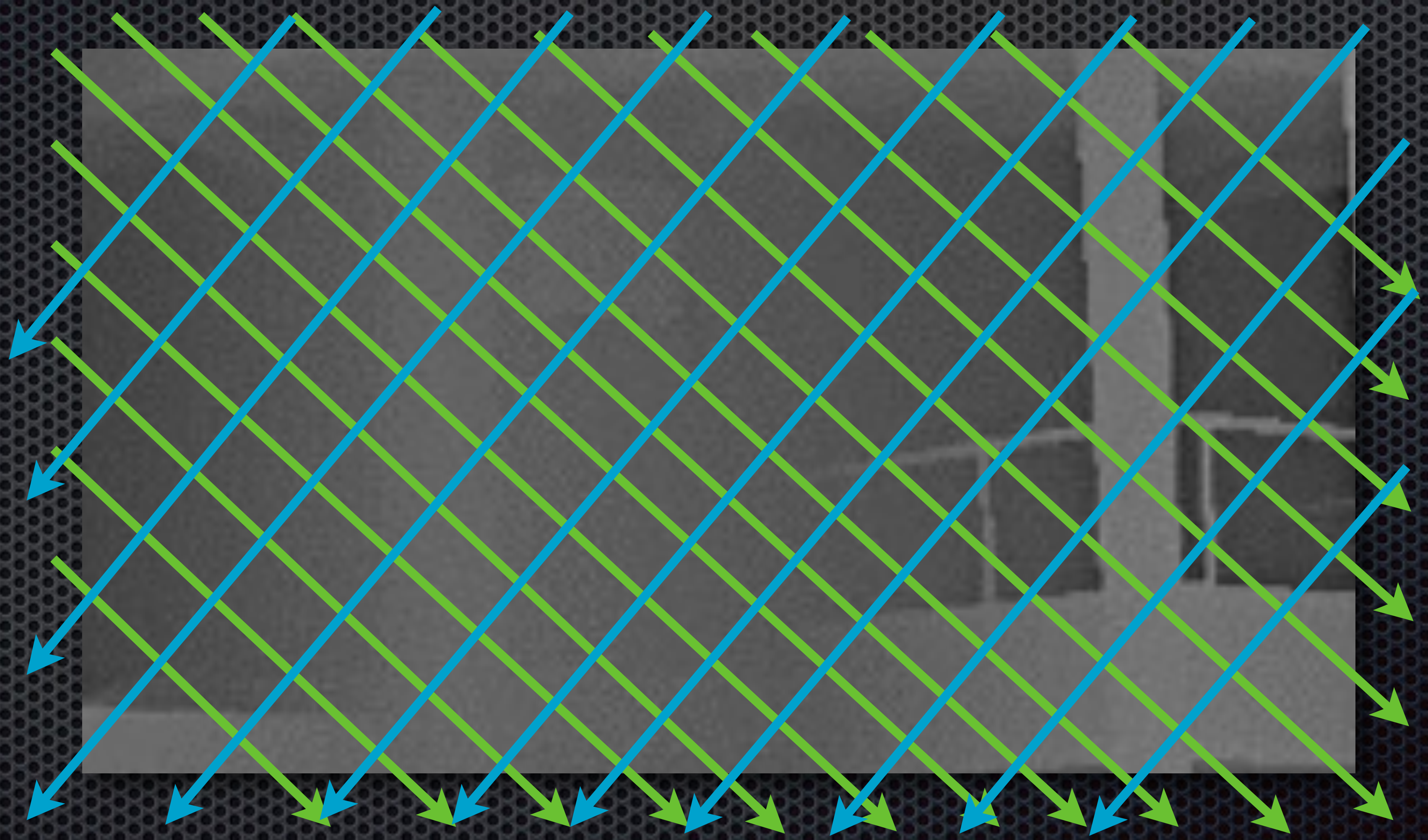


- Amortized constant cost O(1) per pixel per screen direction, unbounded range
- Previous methods, e.g. HBAO, takes multiple (say, 32) iterations for HQ

2 LINE SWEEP AMBIENT OBSCURANCE

Reaching the final result

- Cover the whole framebuffer with lines...

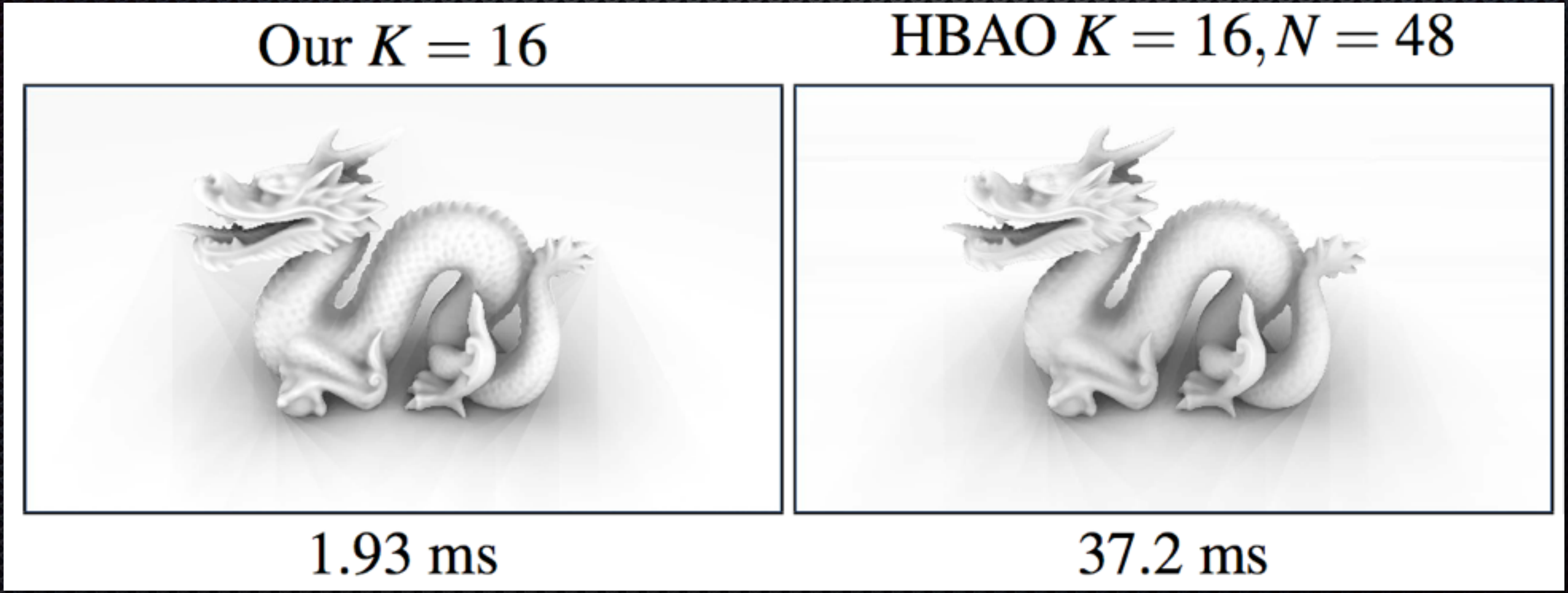


- ...in multiple directions, and gather the results per screen pixel

2 LINE SWEEP AMBIENT OBSCURANCE

1280(+256)x720(+144)

GeForce GTX 480



Our $K = 16$

HBAO $K = 16, N = 48$

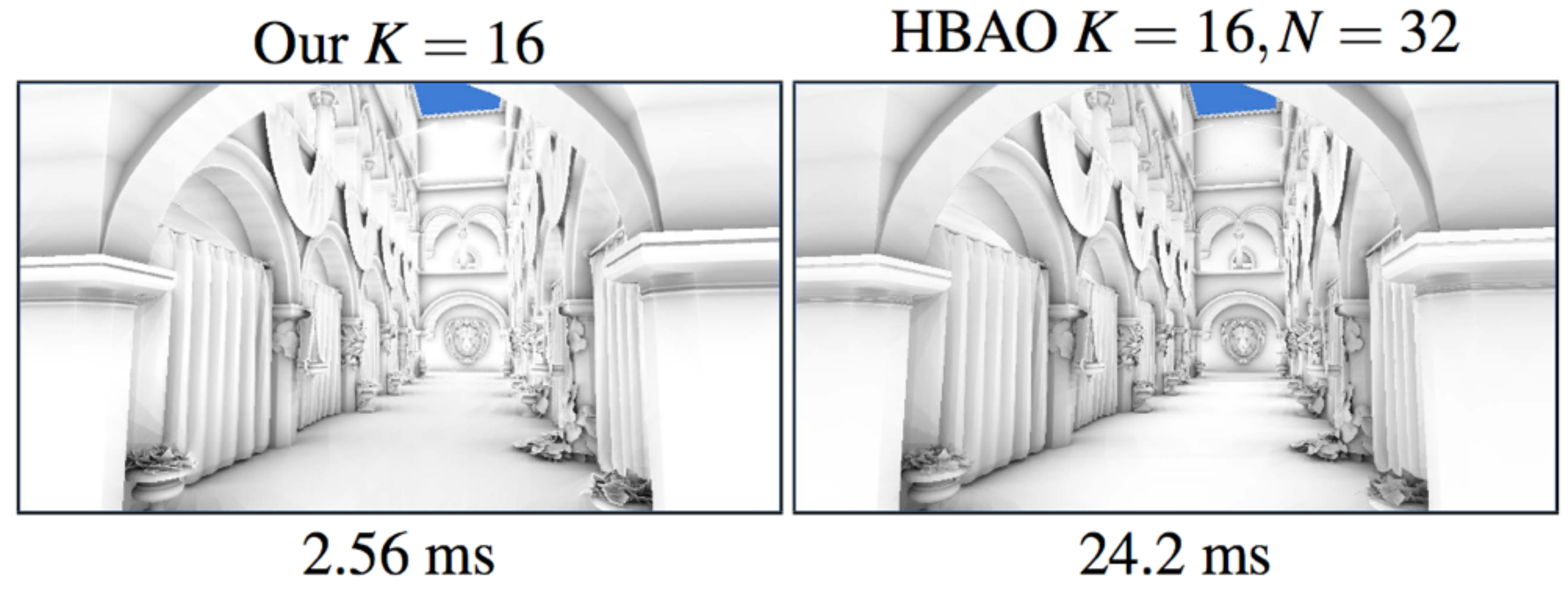
1.93 ms

37.2 ms

2 LINE SWEEP AMBIENT OBSCURANCE

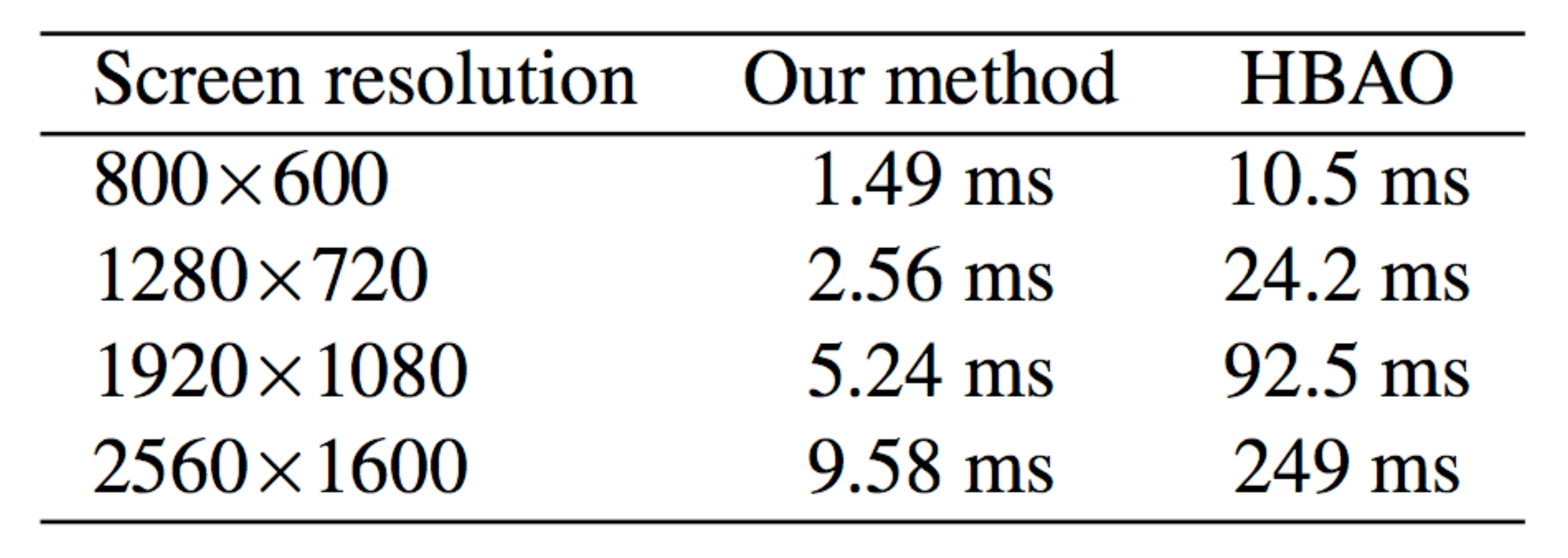
1280(+256)x720(+144)

GeForce GTX 480



2 LINE SWEEP AMBIENT OBSCURANCE

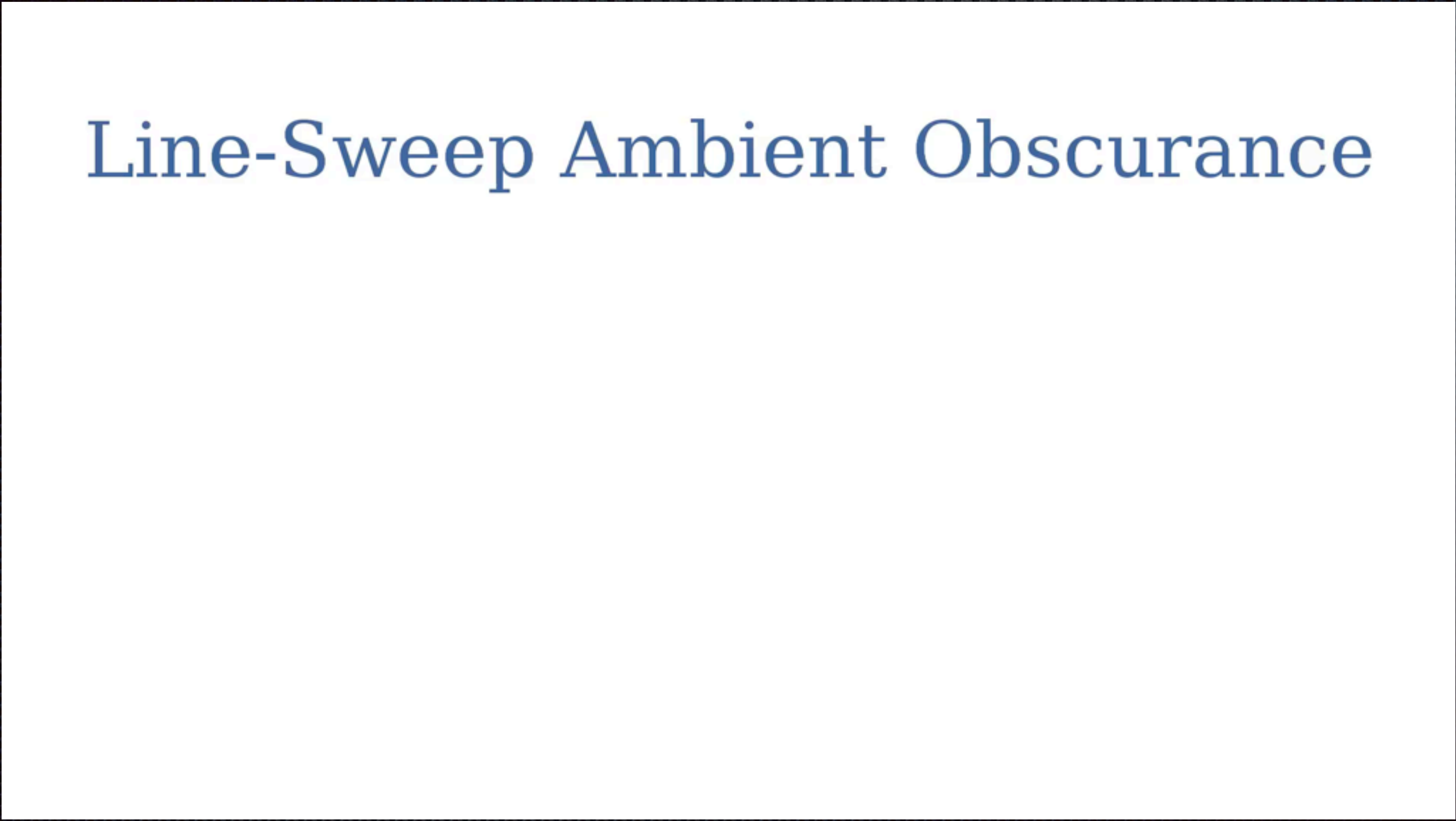
Scaling



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

2 LINE SWEEP AMBIENT OBSCURANCE

VIDEO



Line-Sweep Ambient Obscurance

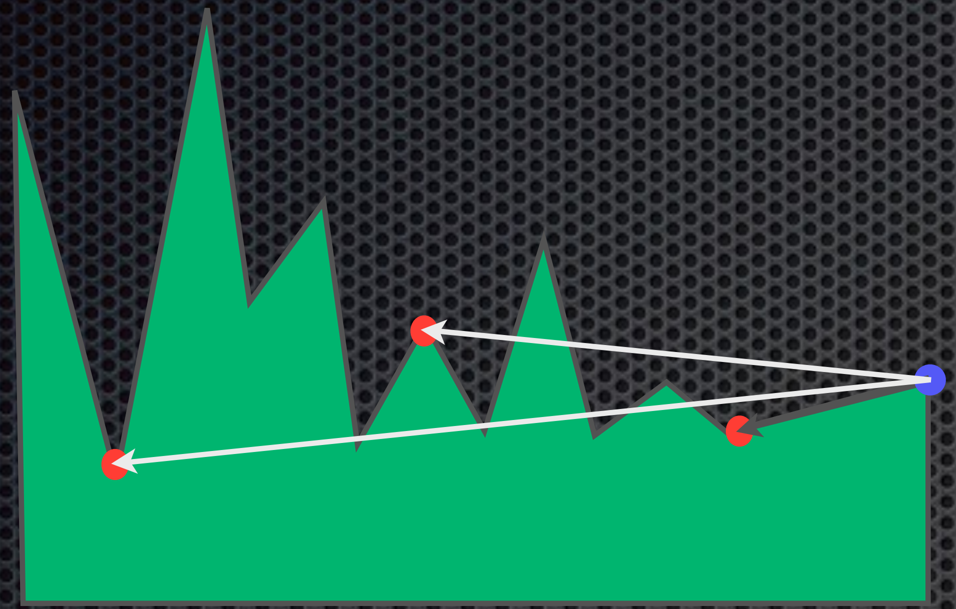
SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

High Performance Graphics 2013
Anaheim, California (ACM)

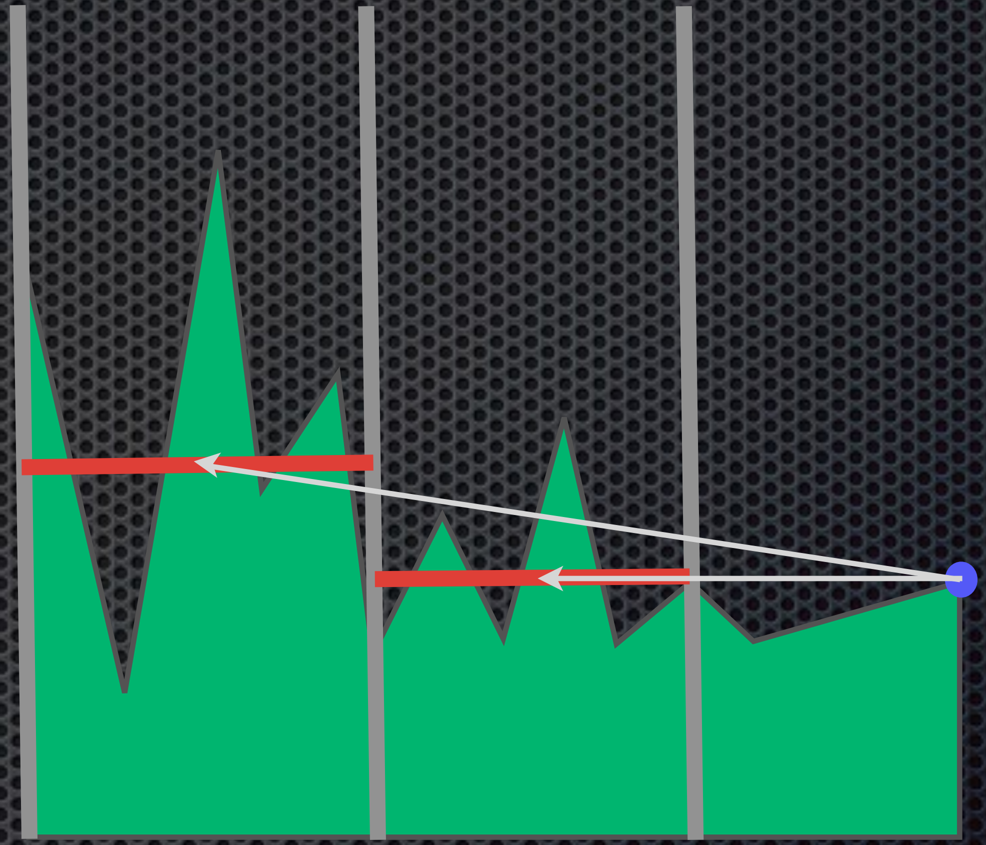
Ville Timonen

3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

Previously used SSAO sampling strategies



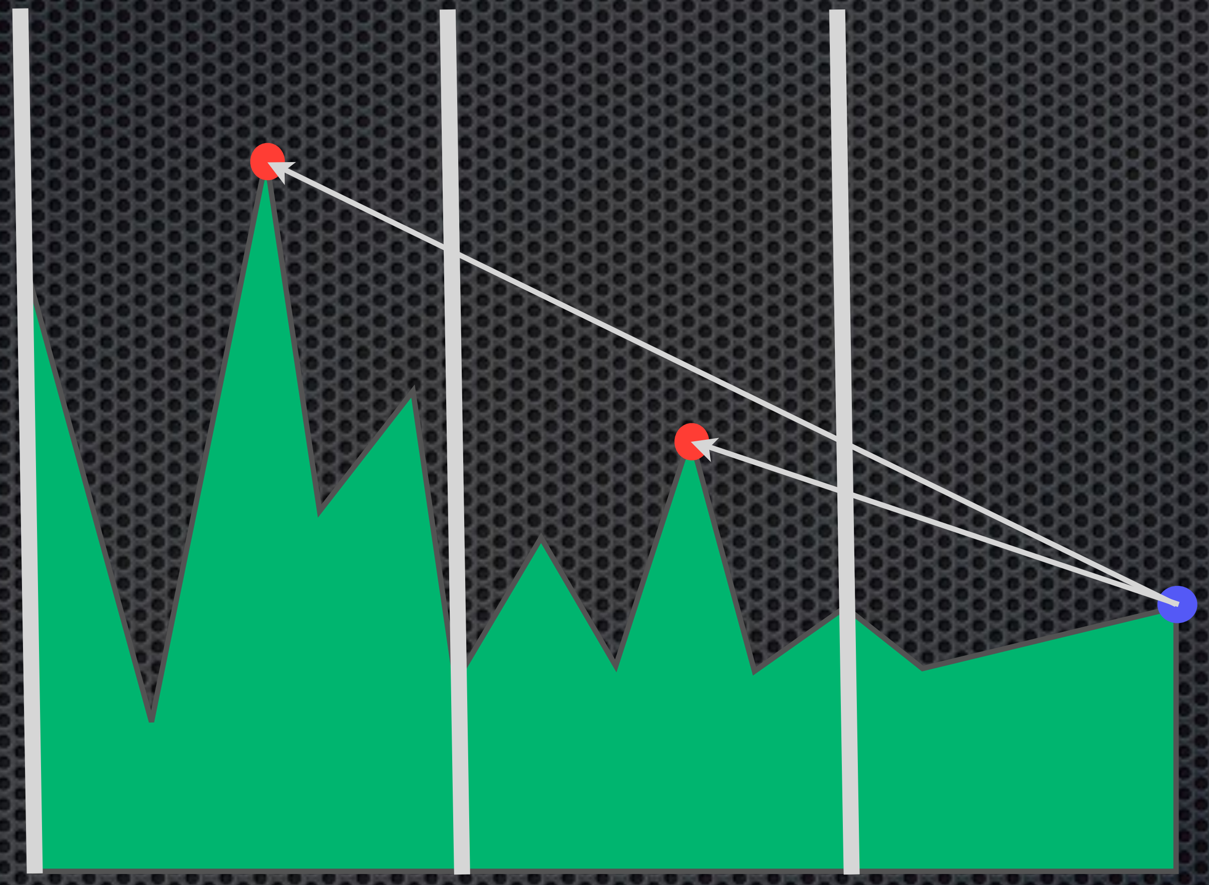
Direct depth buffer samples easily miss important occluders



Mip-mapping flattens the geometry, corrupts silhouettes

3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

Our sampling strategy



We capture points important for AO

3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

Our sampling strategy

1

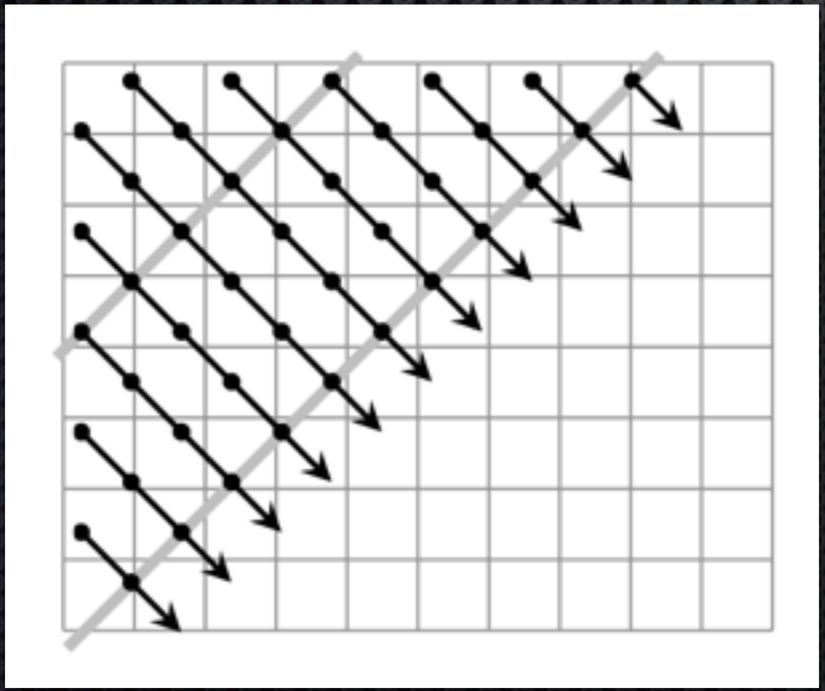
Scans in multiple directions (one direction below)

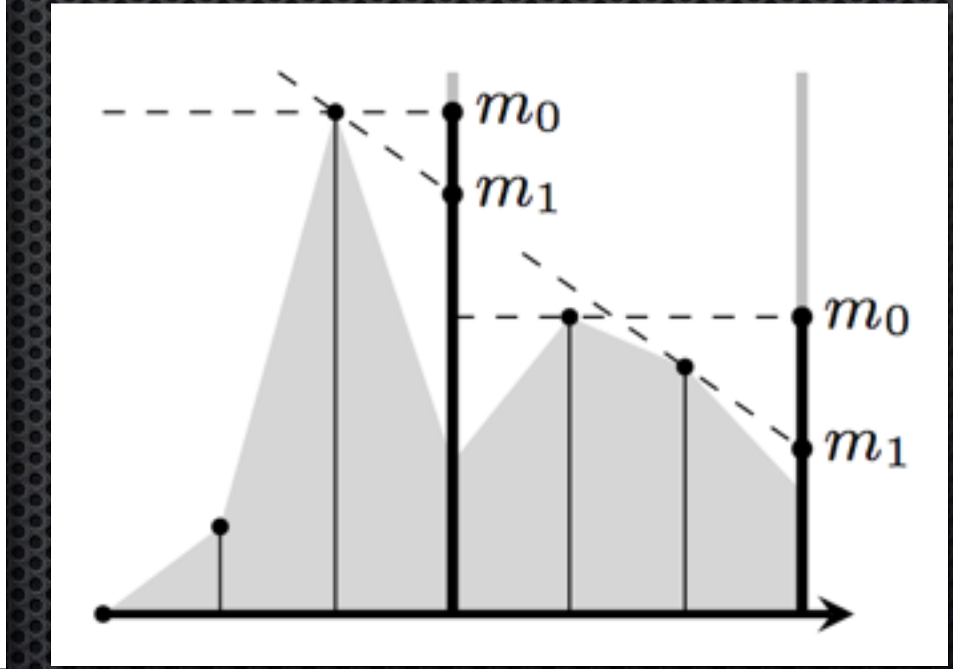
2

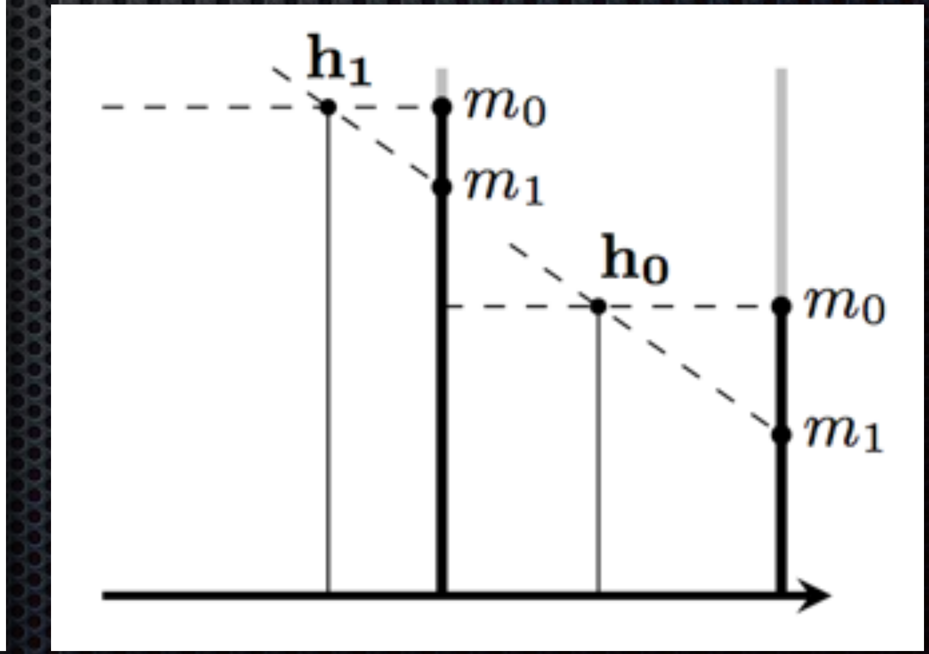
Find highest "projections" (m_0, m_1)

3

Construct final sample points h_i at the intersections

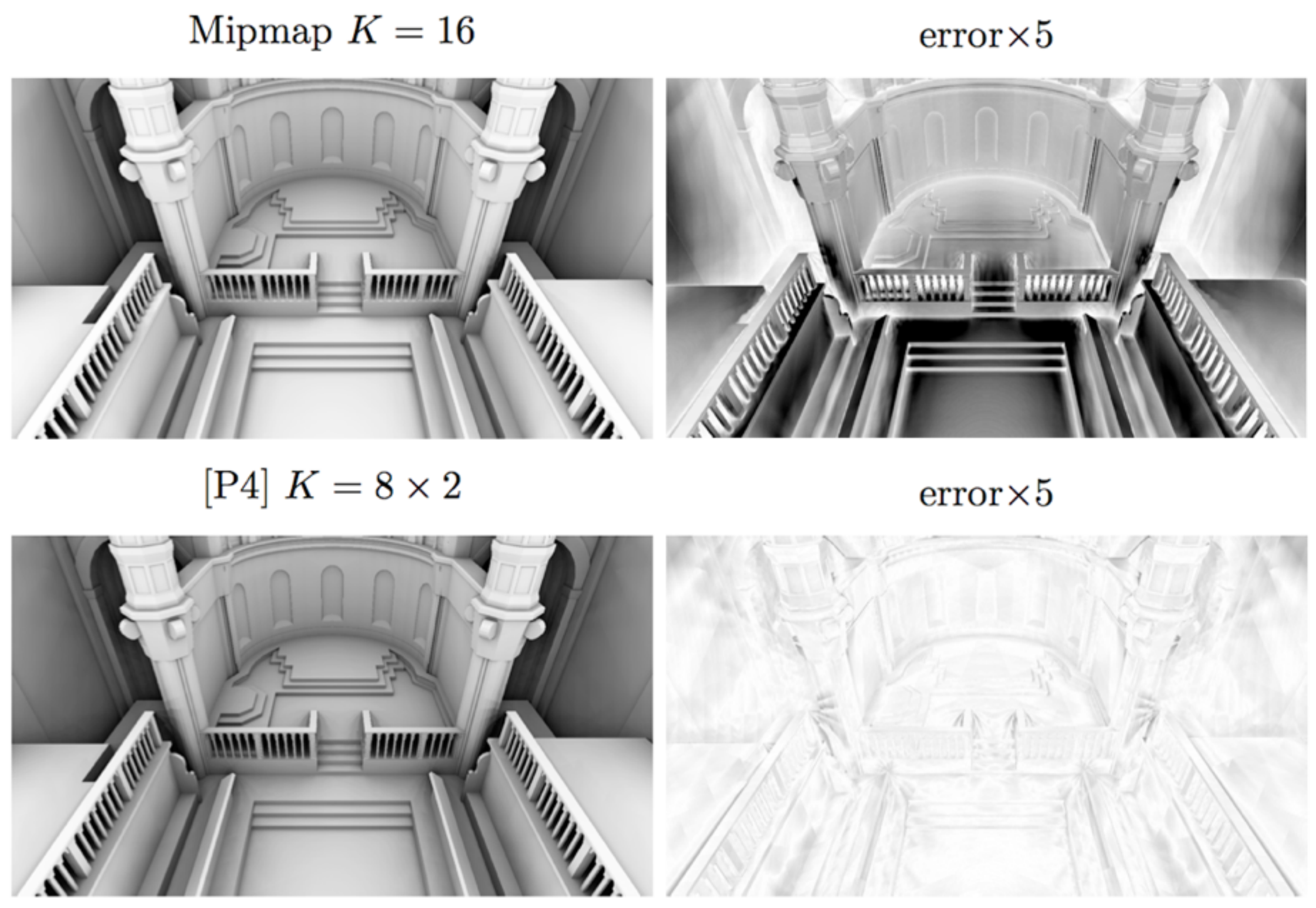






3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

Results



Mipmap $K = 16$

error$\times 5$

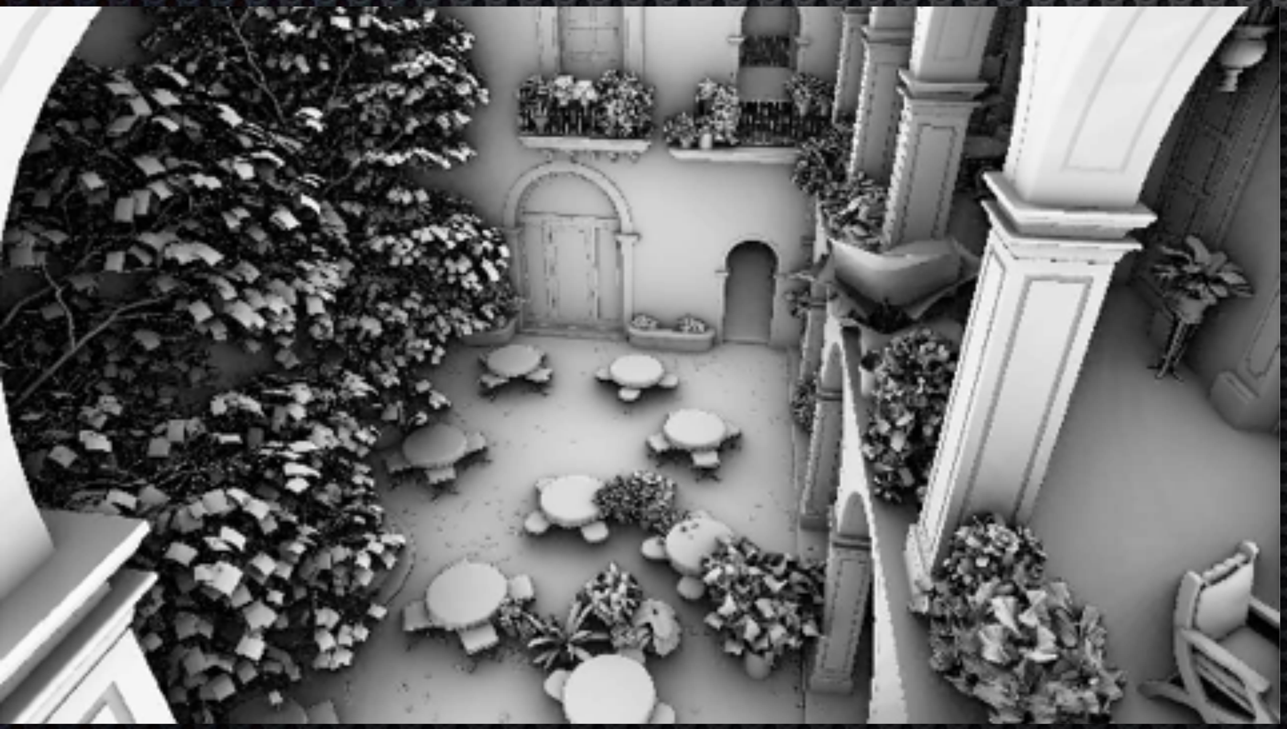
[P4] $K = 8 \times 2$

error$\times 5$

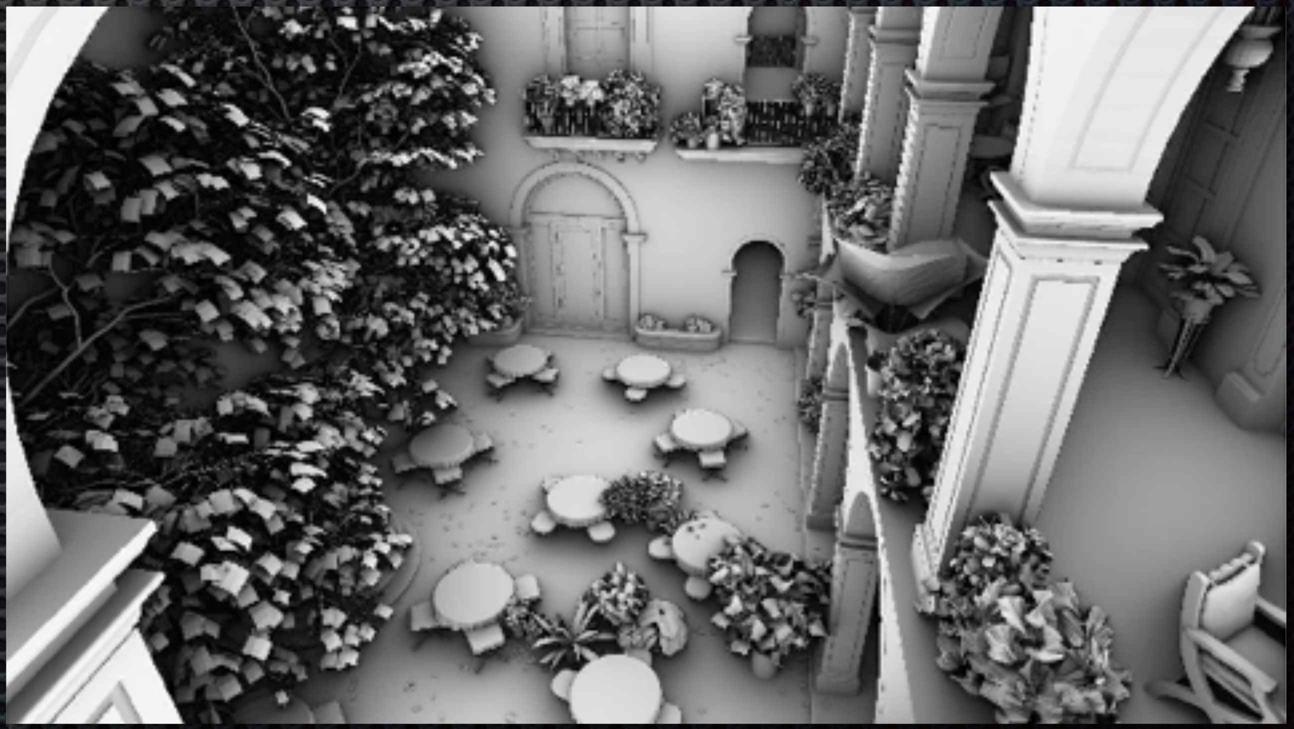
3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

Results

Our (4.6 ms)
Radeon 7970

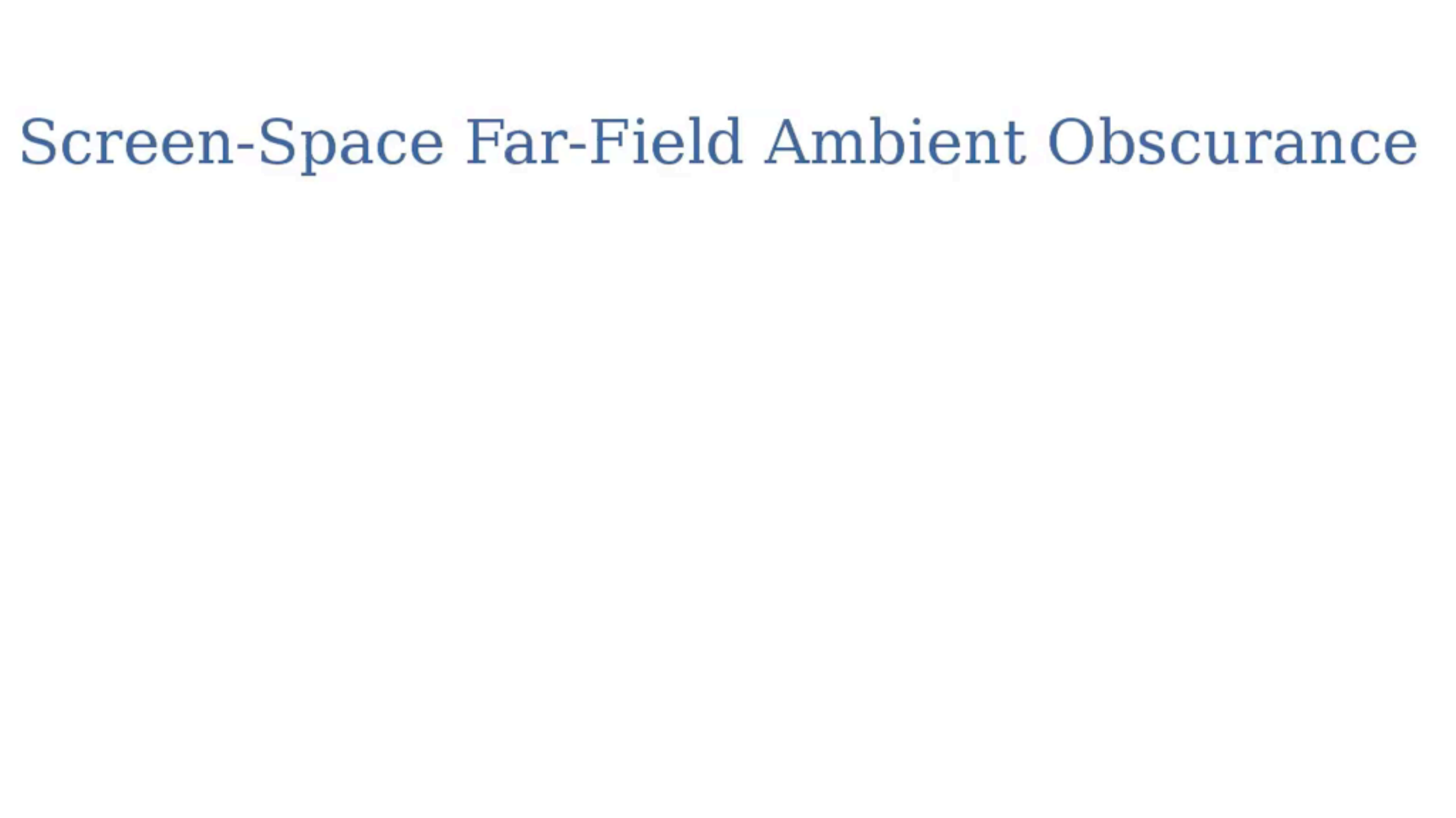


Ray traced (minutes)



3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

VIDEO



Screen-Space Far-Field Ambient Obscurance

SUMMARY

2 LINE SWEEP AMBIENT OBSCURANCE

Reduces time complexity from O(N)* to O(1)

Fast SSAO results without undersampling artefacts

3 SCREEN-SPACE FAR-FIELD AMBIENT OBSCURANCE

Typical multi-res complexity O(log(N))*, but considers the entire framebuffer at real-time framerates

Results very close to a ray traced (screen-space) reference

*) Assuming a NxN framebuffer

5 QUESTIONS

Or comments...

?