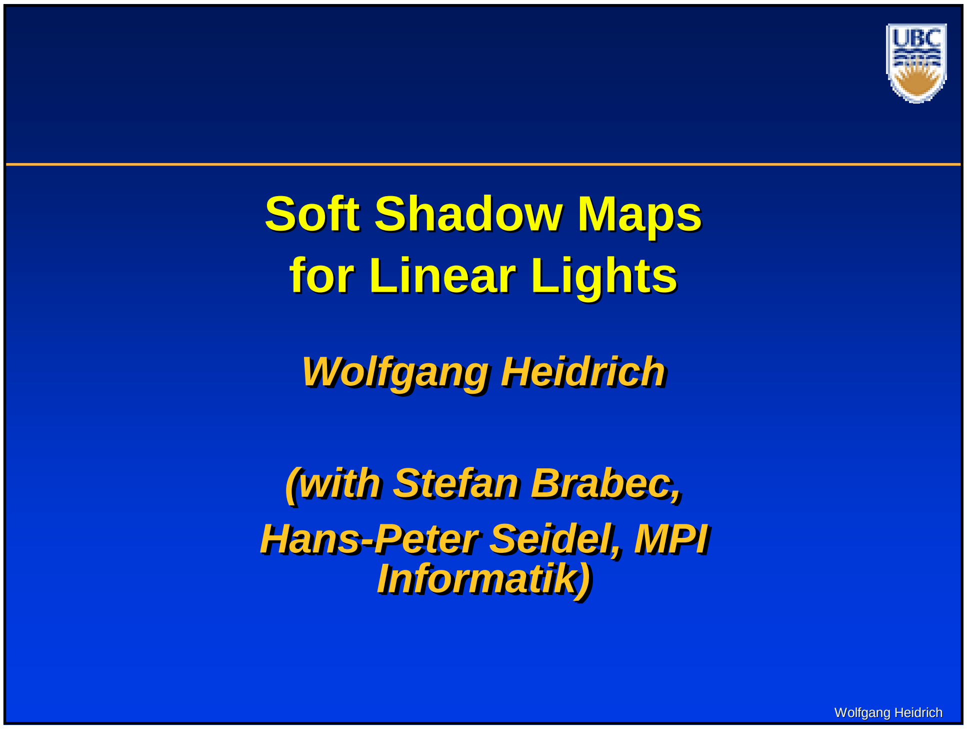
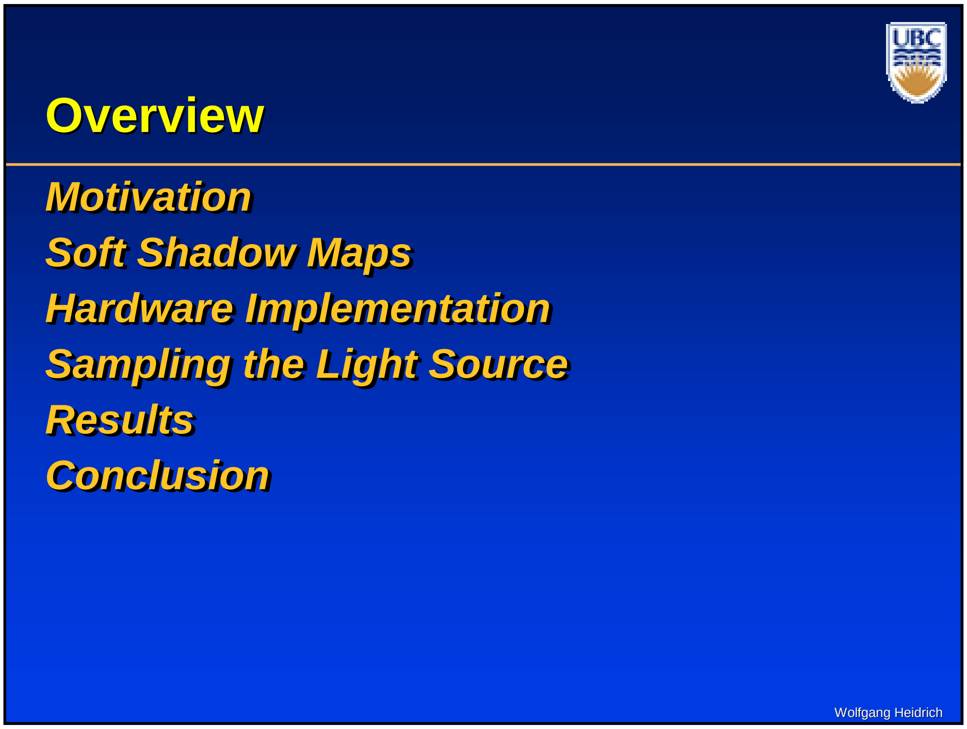


Soft Shadow Maps for Linear Lights

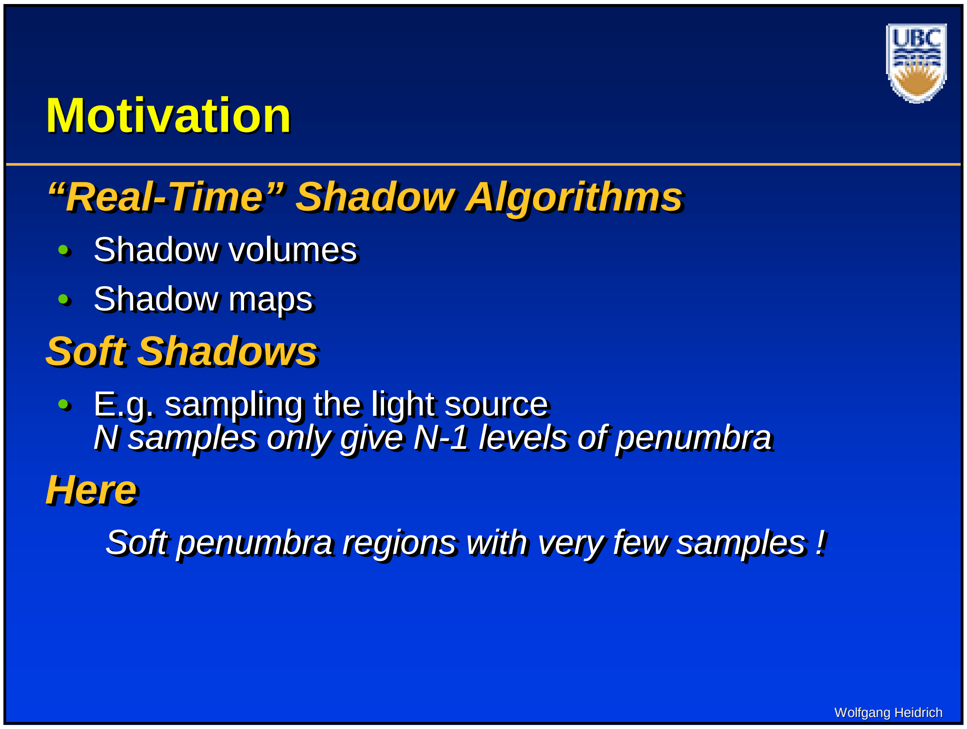
Wolfgang Heidrich

(with Stefan Brabec,
Hans-Peter Seidel, MPI Informatik)



Overview

Motivation
Soft Shadow Maps
Hardware Implementation
Sampling the Light Source
Results
Conclusion



Motivation

“Real-Time” Shadow Algorithms

- Shadow volumes
- Shadow maps

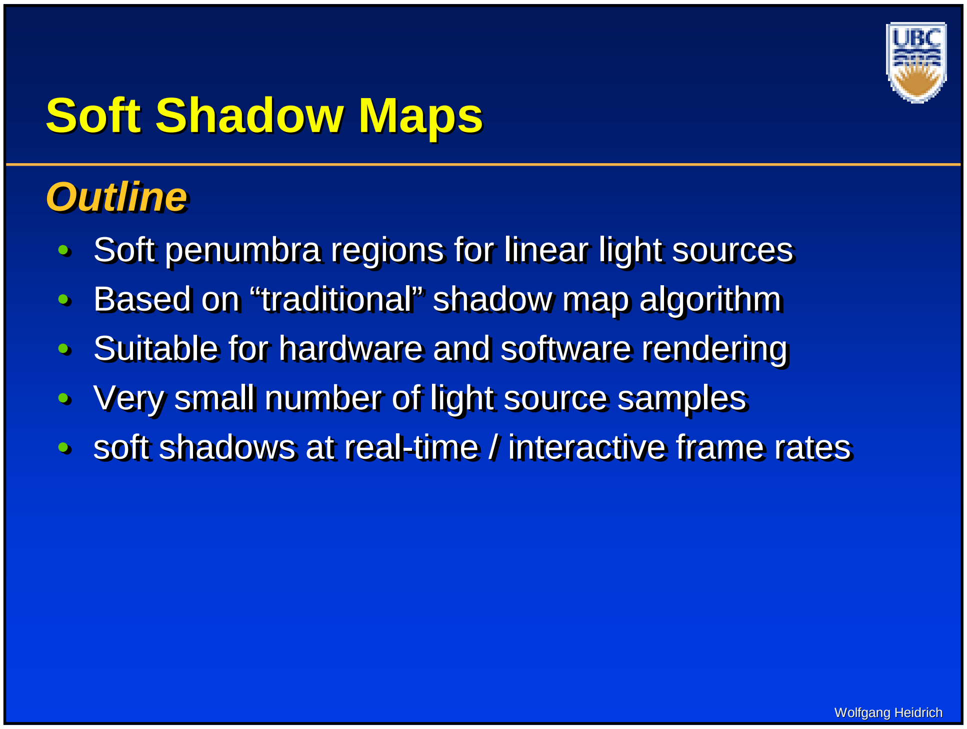
Soft Shadows

- E.g. sampling the light source
  N samples only give N-1 levels of penumbra

Here

Soft penumbra regions with very few samples !

Wolfgang Heidrich

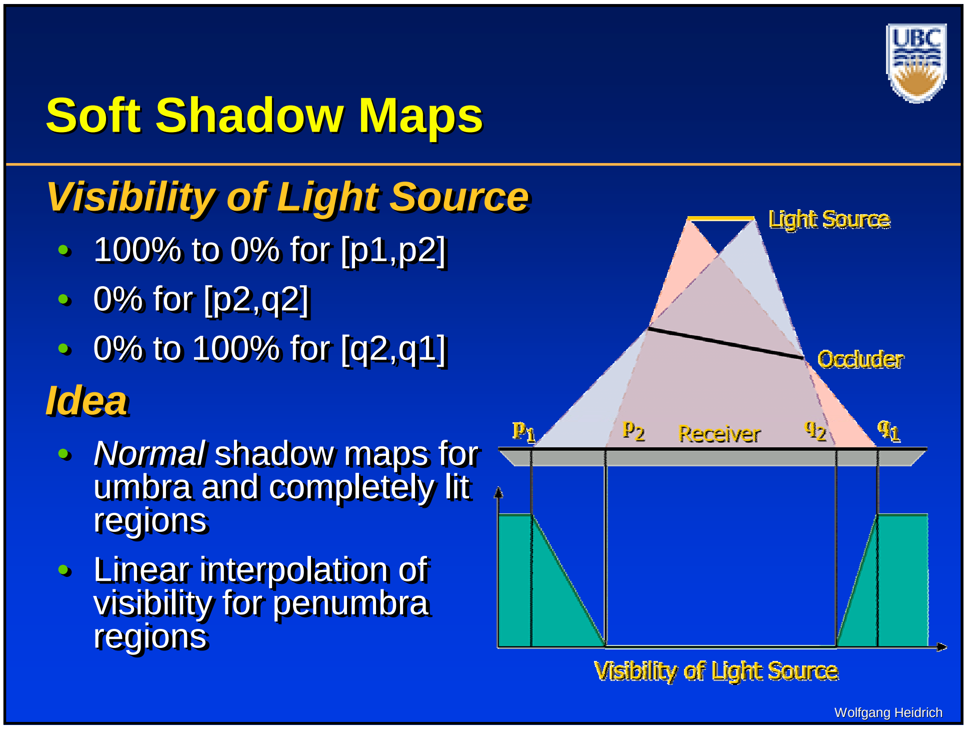


Soft Shadow Maps

Outline

- Soft penumbra regions for linear light sources
- Based on “traditional” shadow map algorithm
- Suitable for hardware and software rendering
- Very small number of light source samples
- soft shadows at real-time / interactive frame rates

Wolfgang Heidrich



Soft Shadow Maps

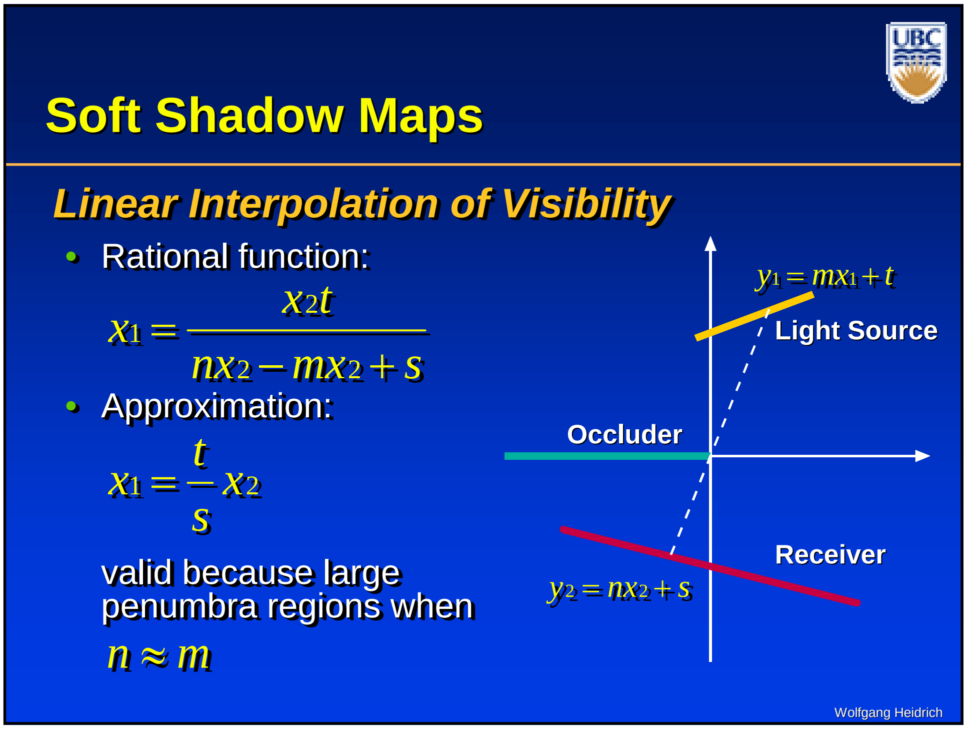
Visibility of Light Source

- 100% to 0% for [p1,p2]
- 0% for [p2,q2]
- 0% to 100% for [q2,q1]

Idea

- *Normal* shadow maps for umbra and completely lit regions
- Linear interpolation of visibility for penumbra regions

Wolfgang Heidrich



Soft Shadow Maps

Linear Interpolation of Visibility

- Rational function:

$$x_1 = \frac{x_2 t}{n x_2 - m x_2 + s}$$

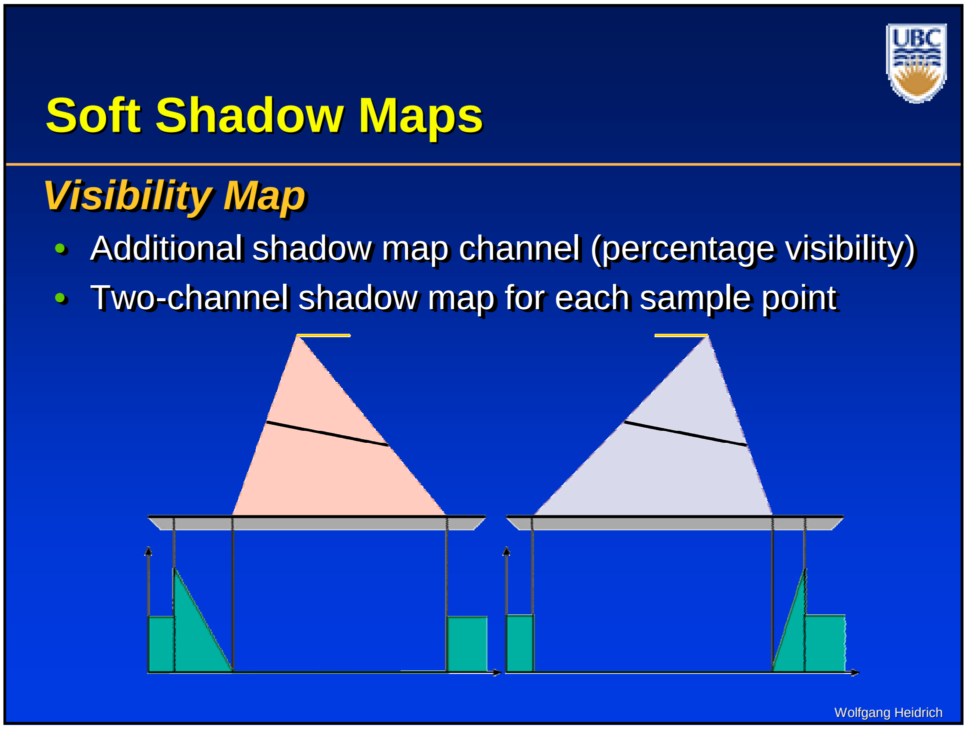
- Approximation:

$$x_1 = \frac{t}{s} x_2$$

valid because large penumbra regions when

$$n \approx m$$

Wolfgang Heidrich

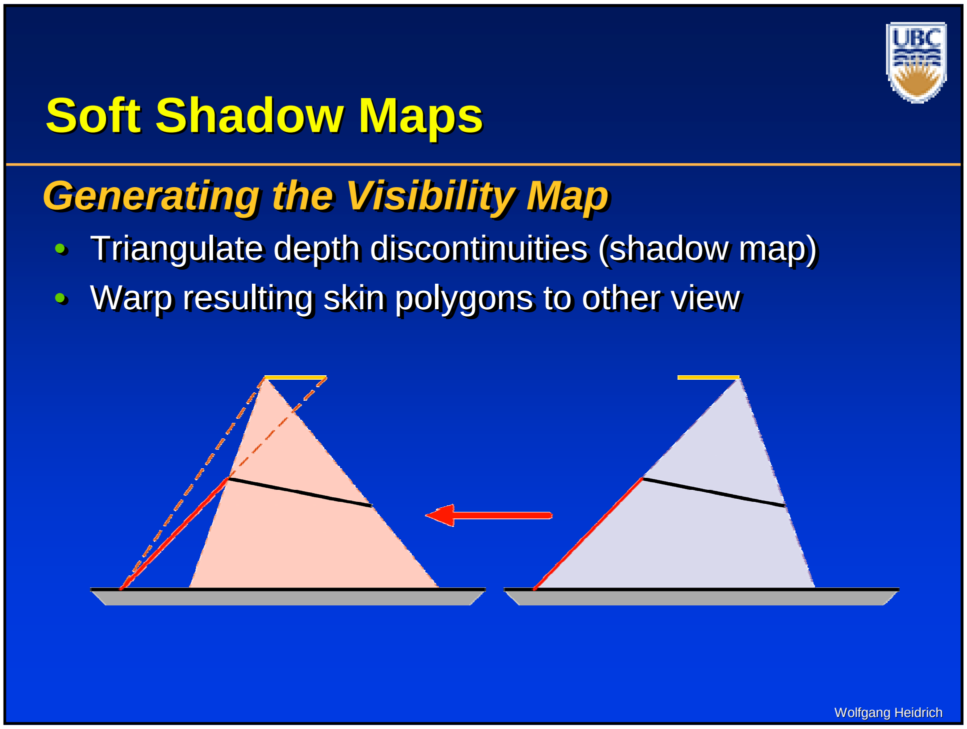


Soft Shadow Maps

Visibility Map

- Additional shadow map channel (percentage visibility)
- Two-channel shadow map for each sample point

Wolfgang Heidrich

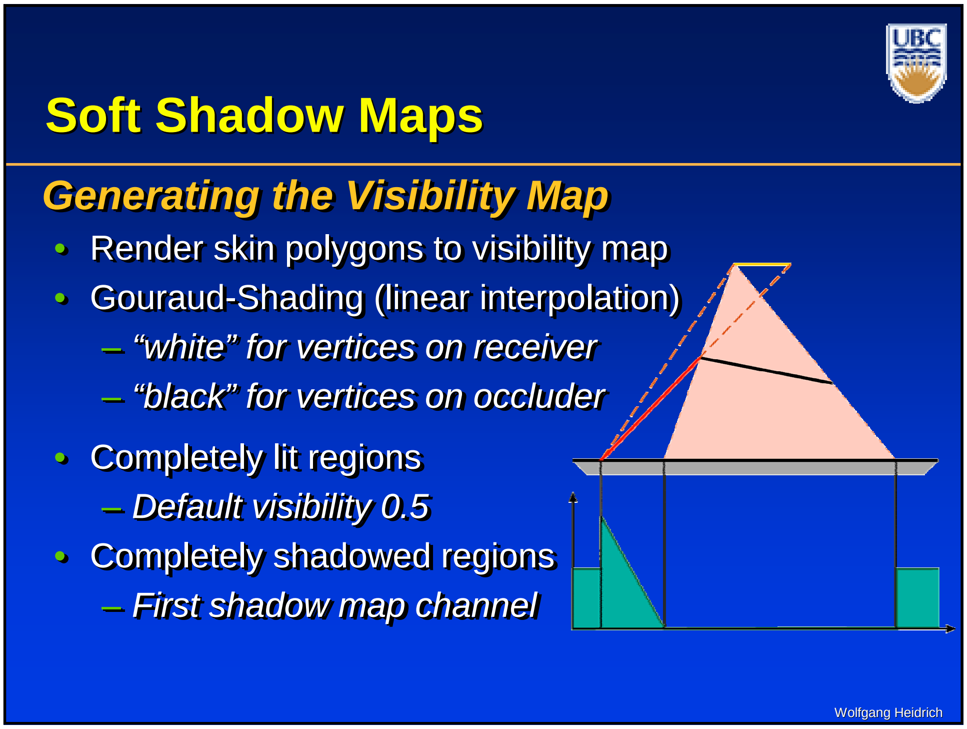


Soft Shadow Maps

Generating the Visibility Map

- Triangulate depth discontinuities (shadow map)
- Warp resulting skin polygons to other view

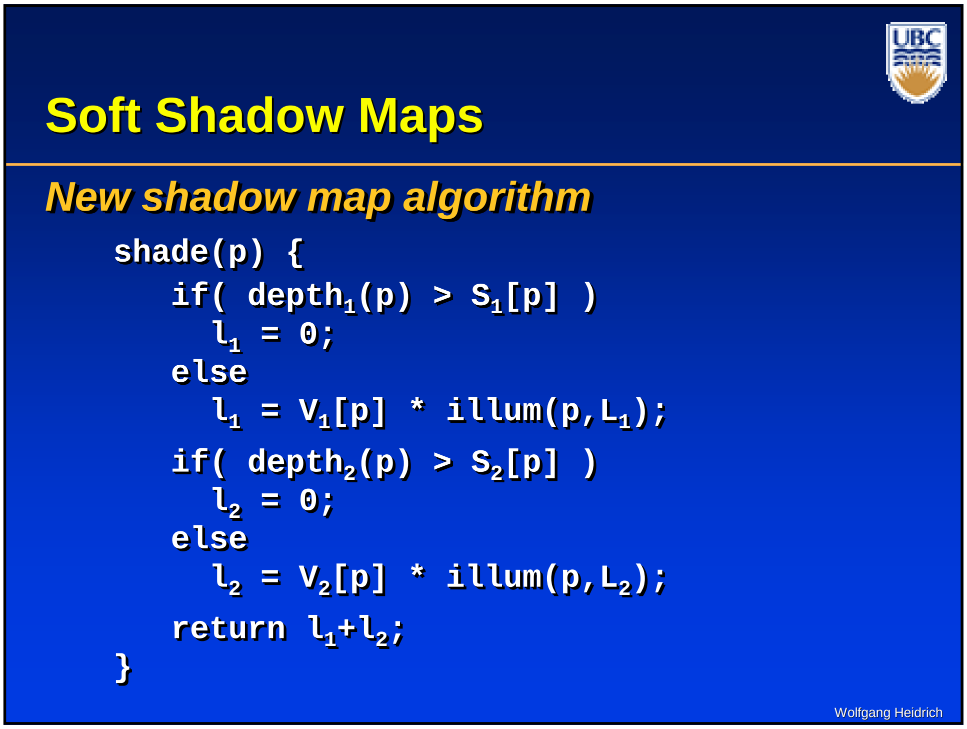
Wolfgang Heidrich



Soft Shadow Maps

Generating the Visibility Map

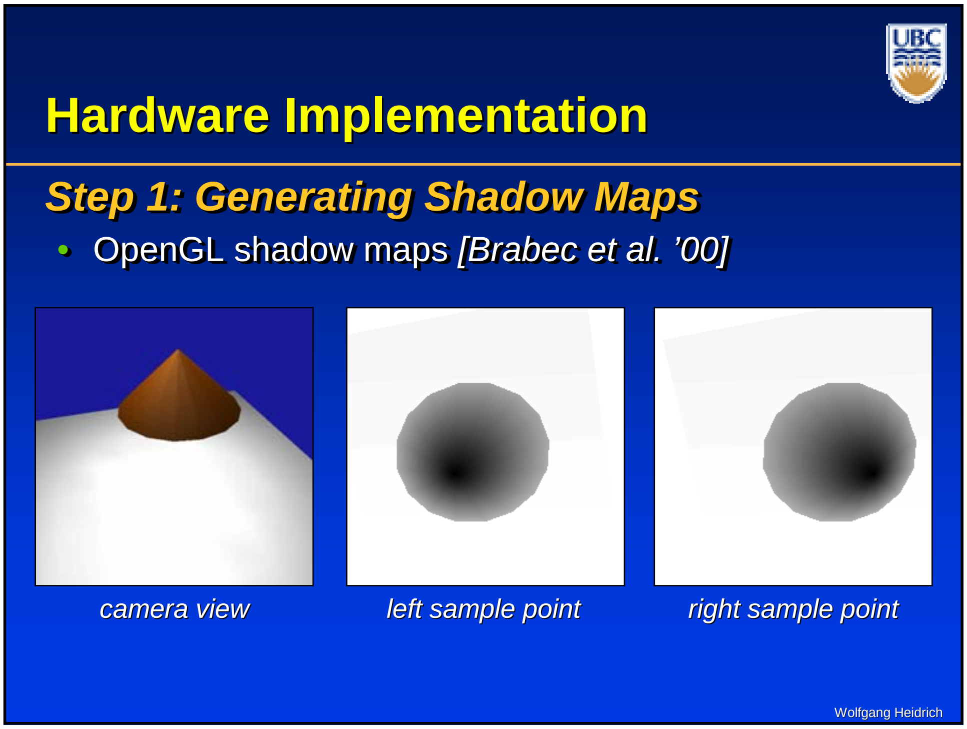
- Render skin polygons to visibility map
- Gouraud-Shading (linear interpolation)
  - *"white" for vertices on receiver*
  - *"black" for vertices on occluder*
- Completely lit regions
  - *Default visibility 0.5*
- Completely shadowed regions
  - *First shadow map channel*



Soft Shadow Maps

New shadow map algorithm

```
shade(p) {
   if( depth_1(p) > S_1[p] )
     l_1 = 0;
   else
     l_1 = V_1[p] * illum(p,L_1);
   if( depth_2(p) > S_2[p] )
     l_2 = 0;
   else
     l_2 = V_2[p] * illum(p,L_2);
   return l_1+l_2;
}
```



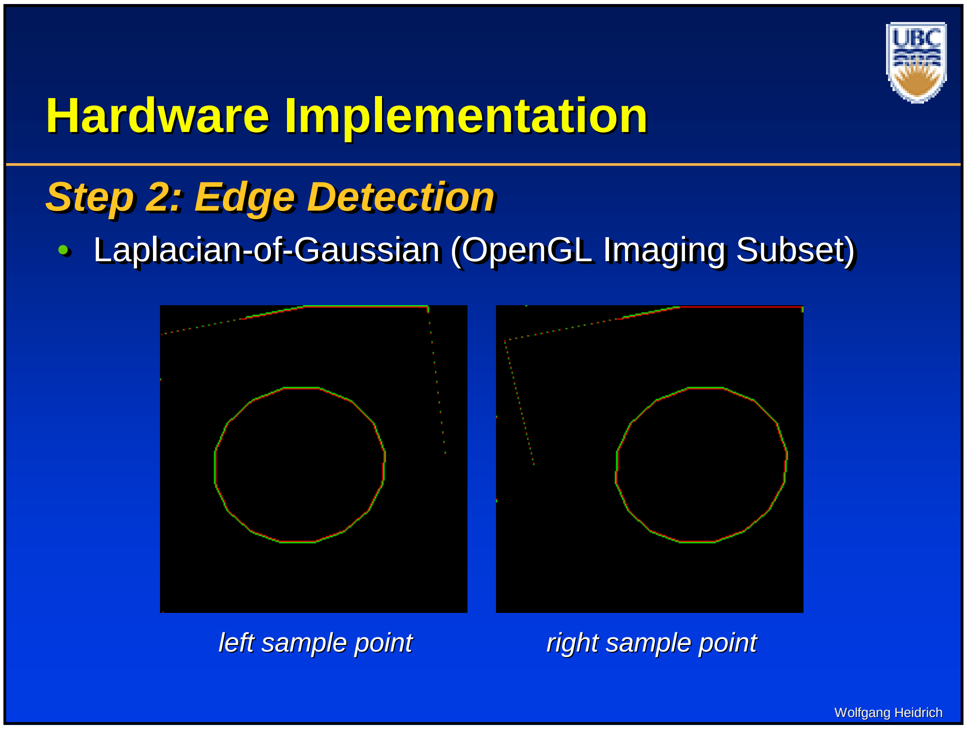
Hardware Implementation

Step 1: Generating Shadow Maps

- OpenGL shadow maps *[Brabec et al. ’00]*

camera view *left sample point* *right sample point*

Wolfgang Heidrich



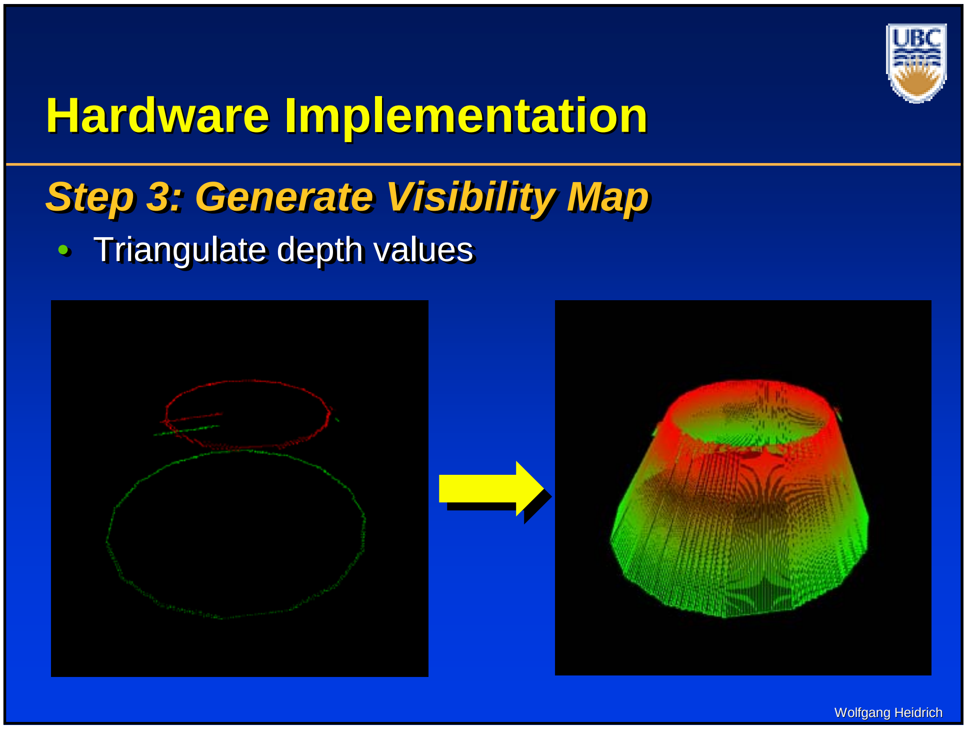
Hardware Implementation

Step 2: Edge Detection

- Laplacian-of-Gaussian (OpenGL Imaging Subset)

left sample point *right sample point*

Wolfgang Heidrich

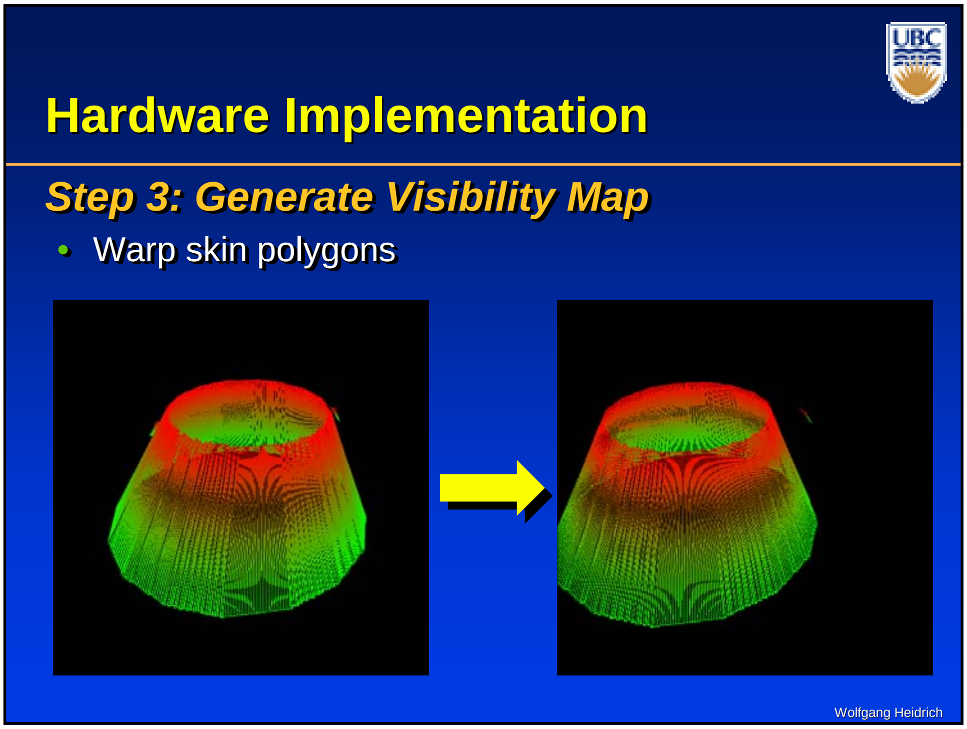


Hardware Implementation

Step 3: Generate Visibility Map

- Triangulate depth values

Wolfgang Heidrich

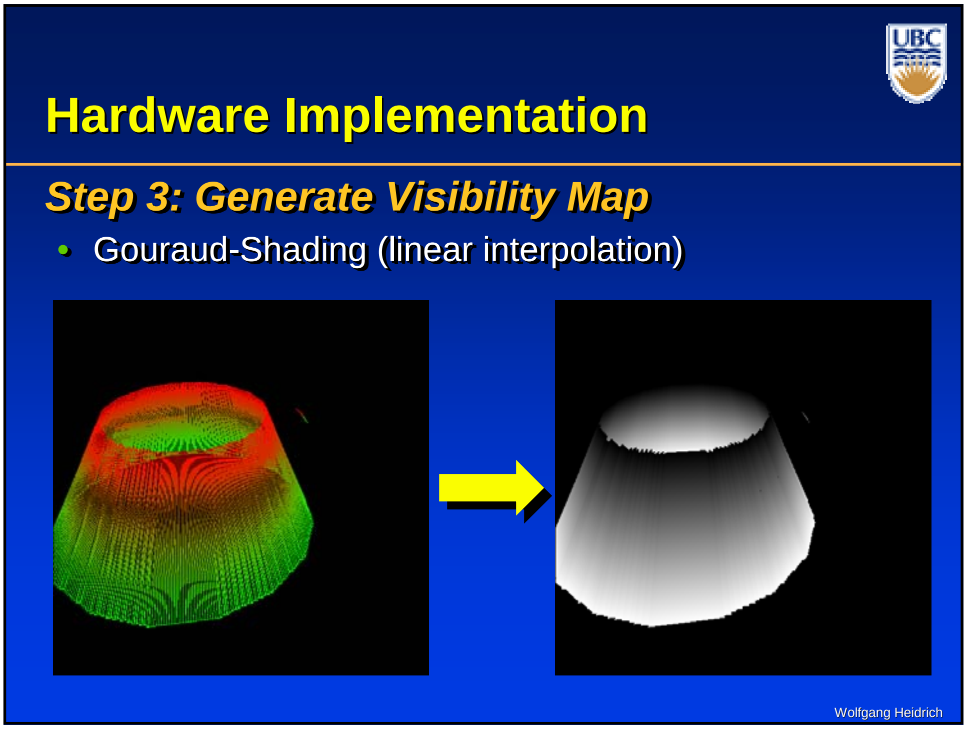


Hardware Implementation

Step 3: Generate Visibility Map

- Warp skin polygons

Wolfgang Heidrich

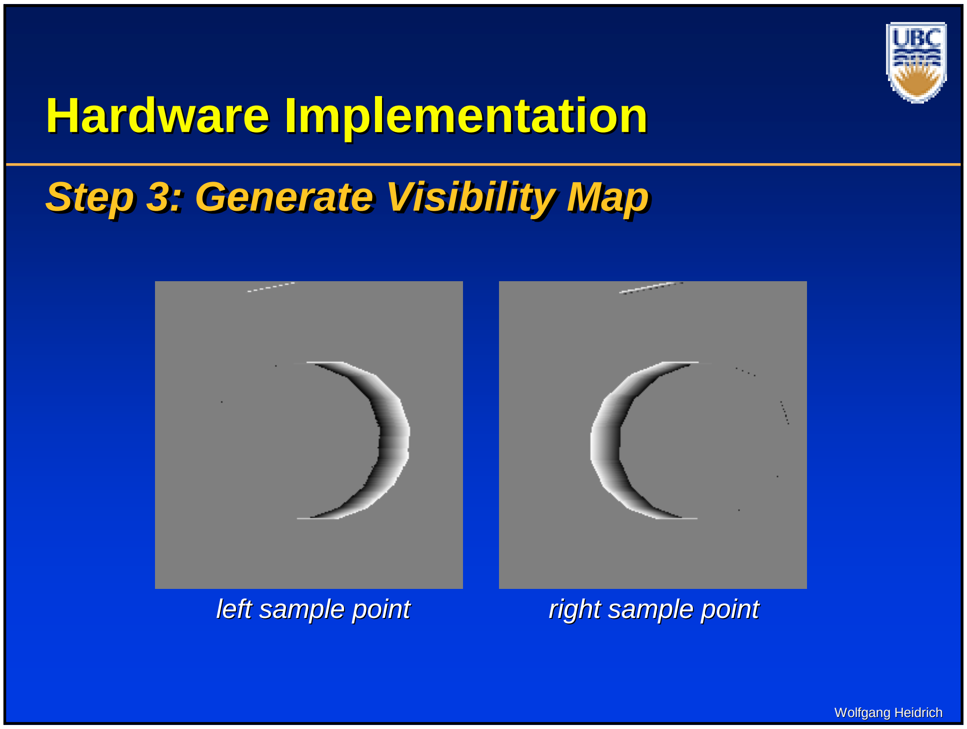


Hardware Implementation

Step 3: Generate Visibility Map

- Gouraud-Shading (linear interpolation)

Wolfgang Heidrich



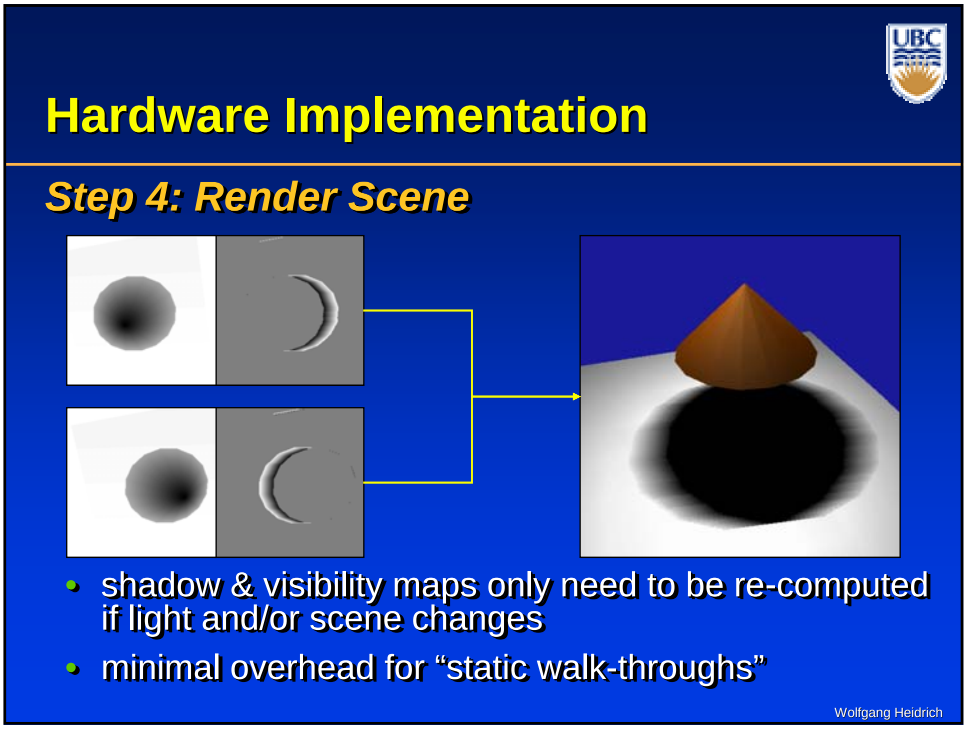
Hardware Implementation

Step 3: Generate Visibility Map

left sample point

right sample point

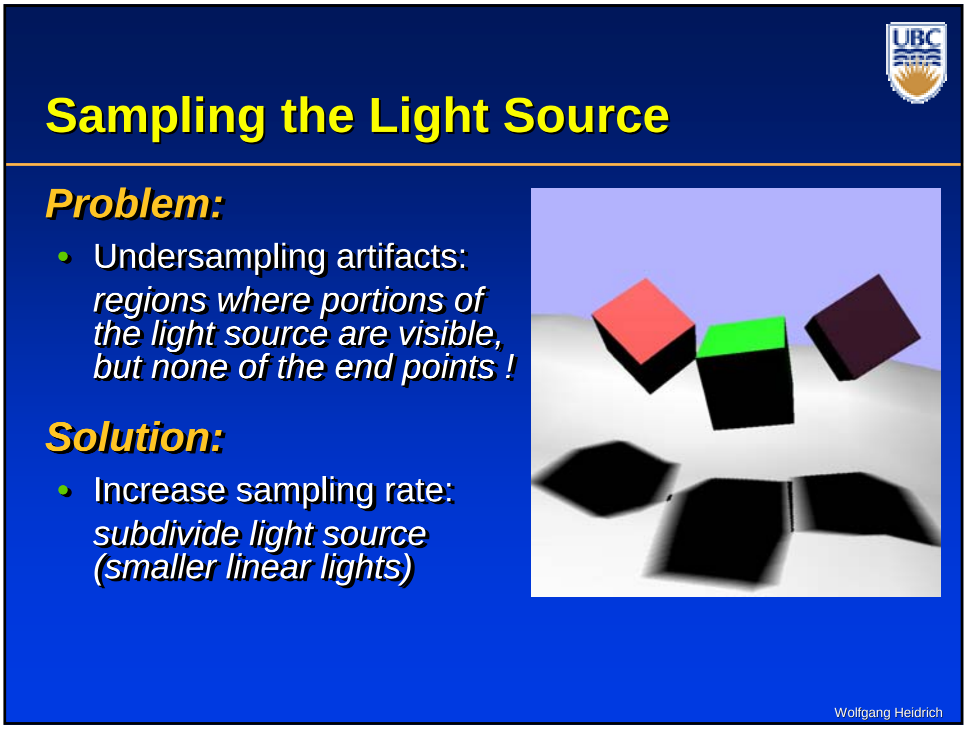
Wolfgang Heidrich



Hardware Implementation

Step 4: Render Scene

- shadow & visibility maps only need to be re-computed if light and/or scene changes
- minimal overhead for “static walk-throughs”



Sampling the Light Source

Problem:

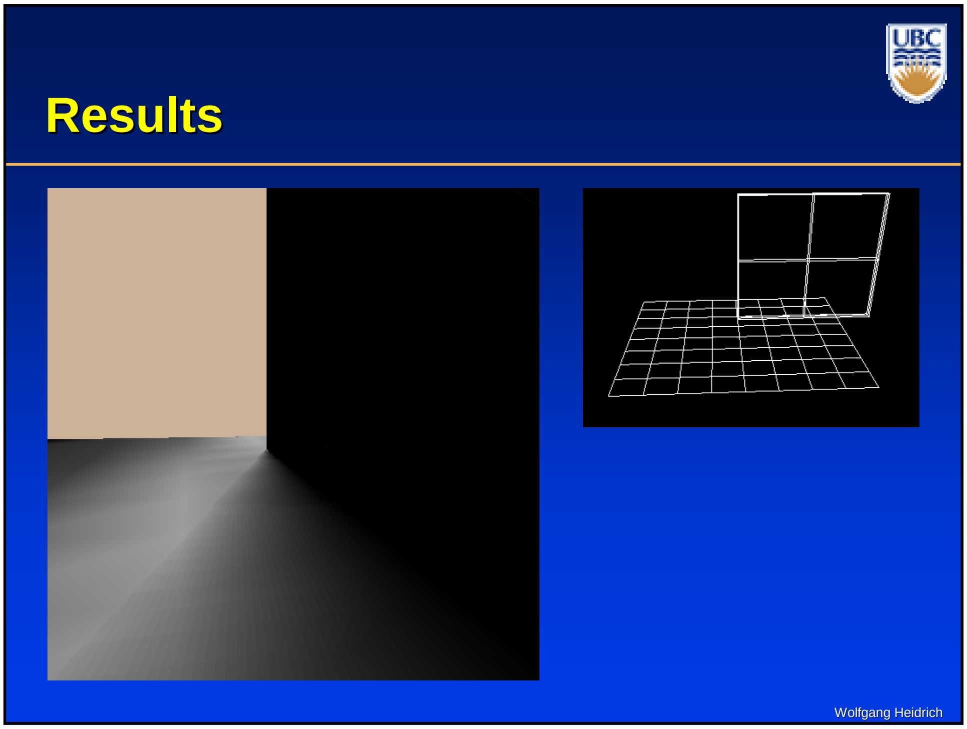
- Undersampling artifacts:

  regions where portions of the light source are visible, but none of the end points !

Solution:

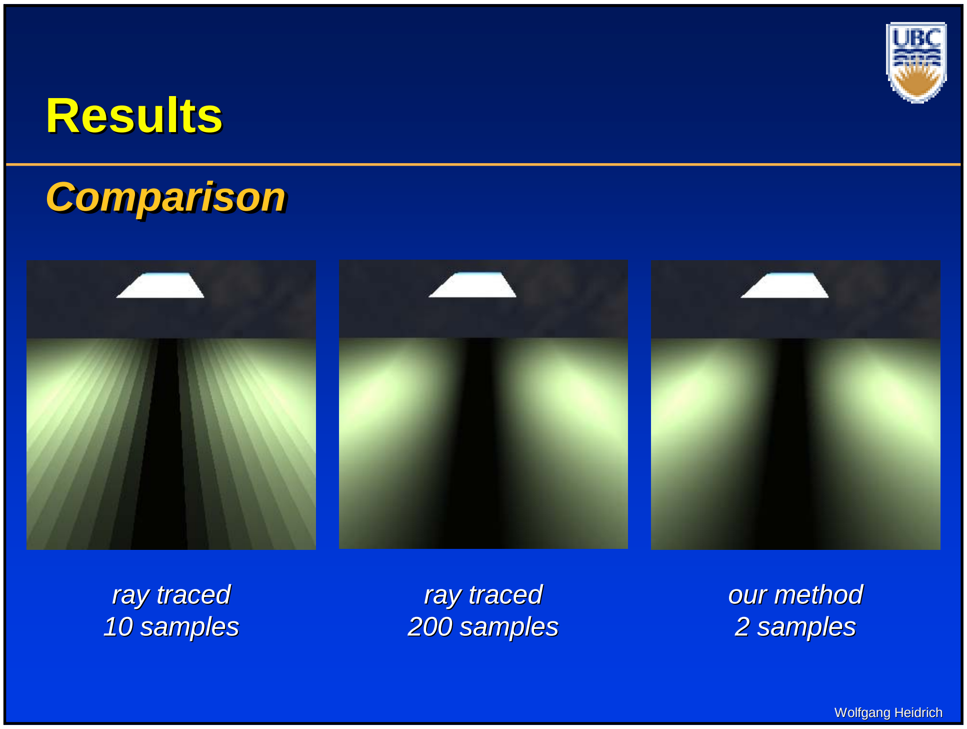
- Increase sampling rate:

  subdivide light source (smaller linear lights)



Results

Wolfgang Heidrich



Results

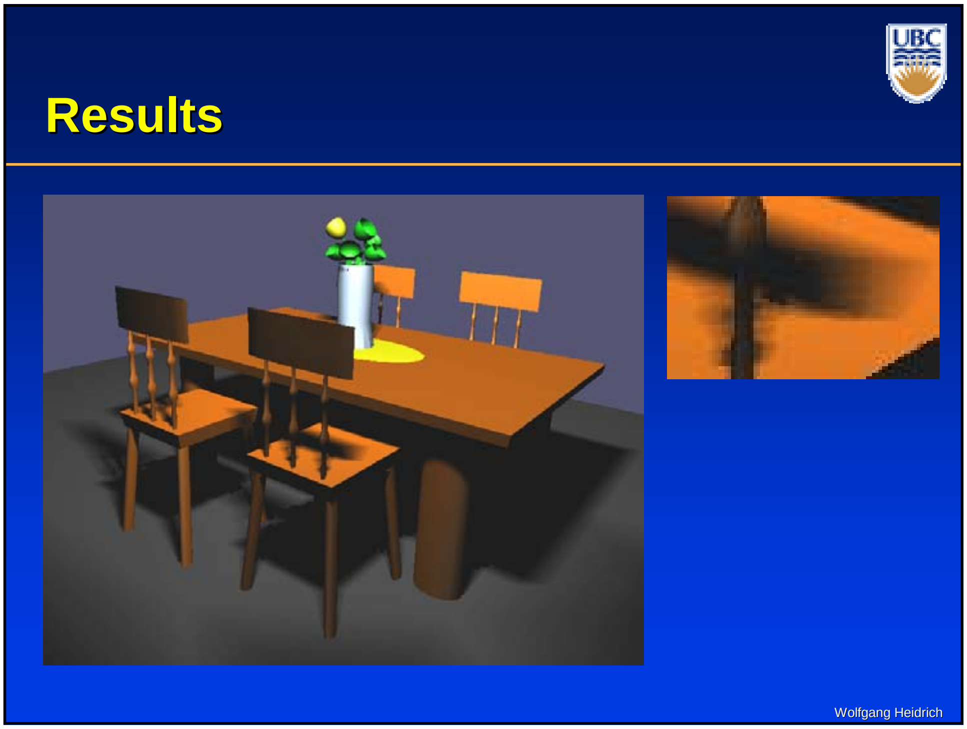
Comparison

ray traced
10 samples

ray traced
200 samples

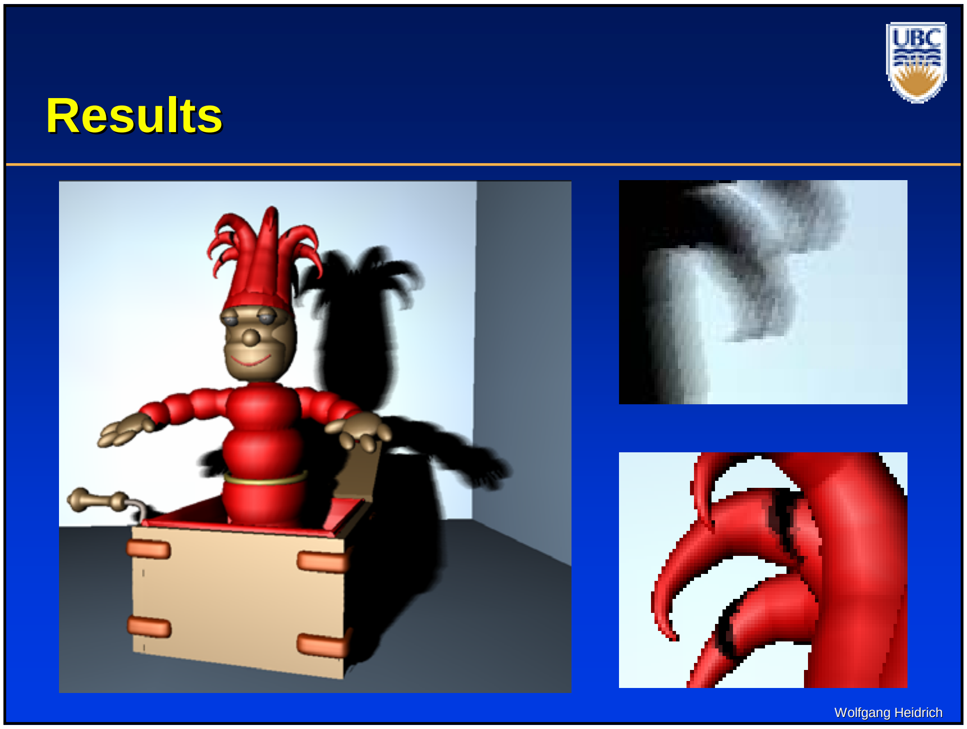
our method
2 samples

Wolfgang Heidrich



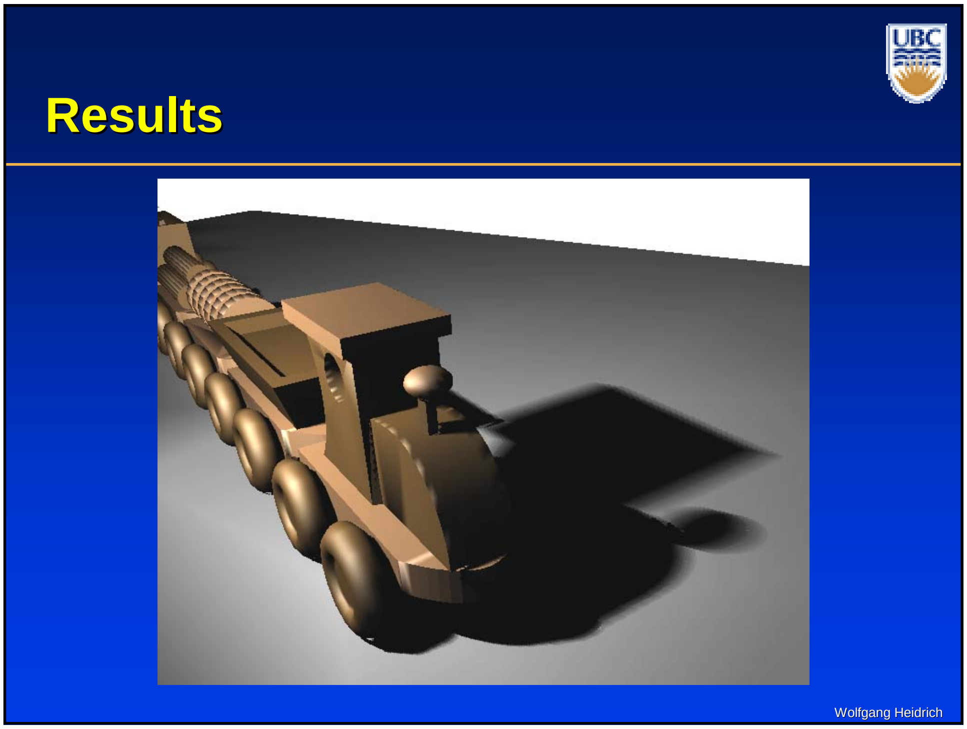
Results

Wolfgang Heidrich

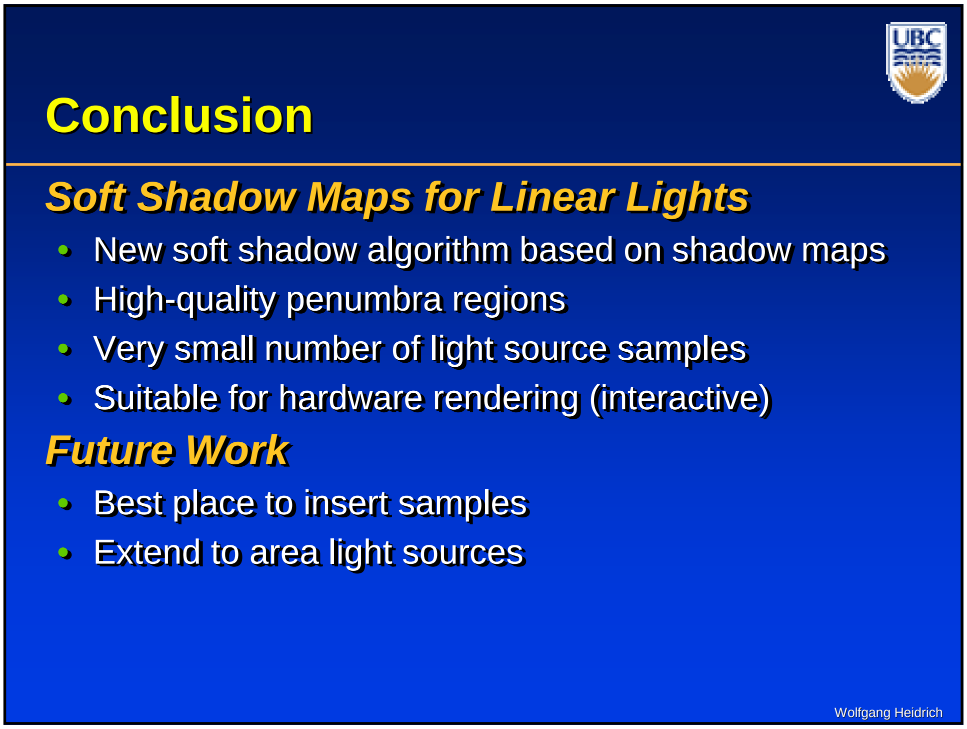


Results

Wolfgang Heidrich



Results



Conclusion

Soft Shadow Maps for Linear Lights

- New soft shadow algorithm based on shadow maps
- High-quality penumbra regions
- Very small number of light source samples
- Suitable for hardware rendering (interactive)

Future Work

- Best place to insert samples
- Extend to area light sources