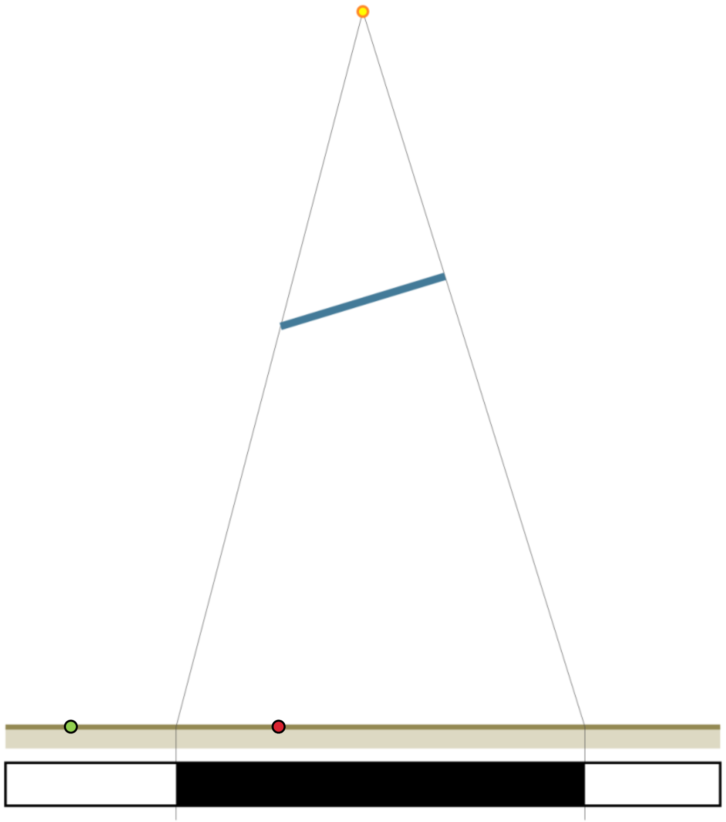
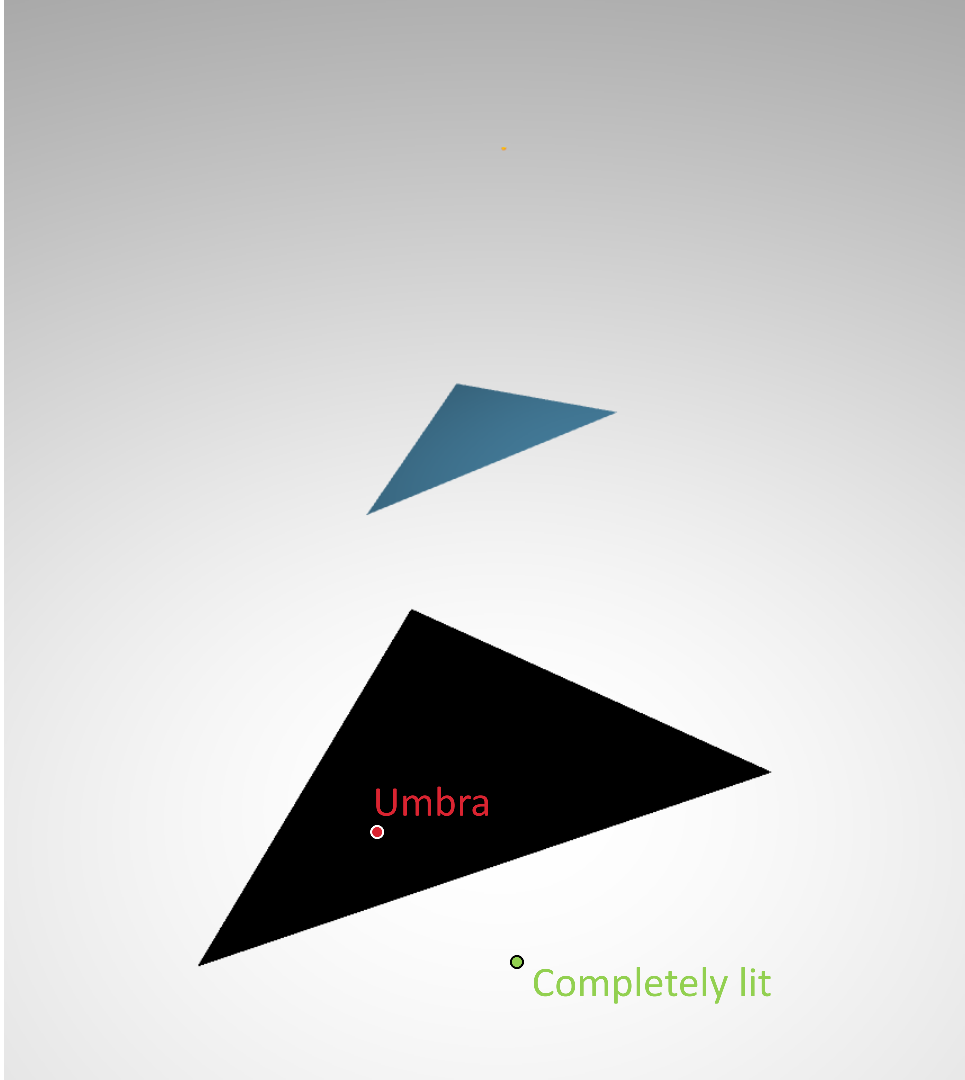
Soft Shadows

Michael Schwarz

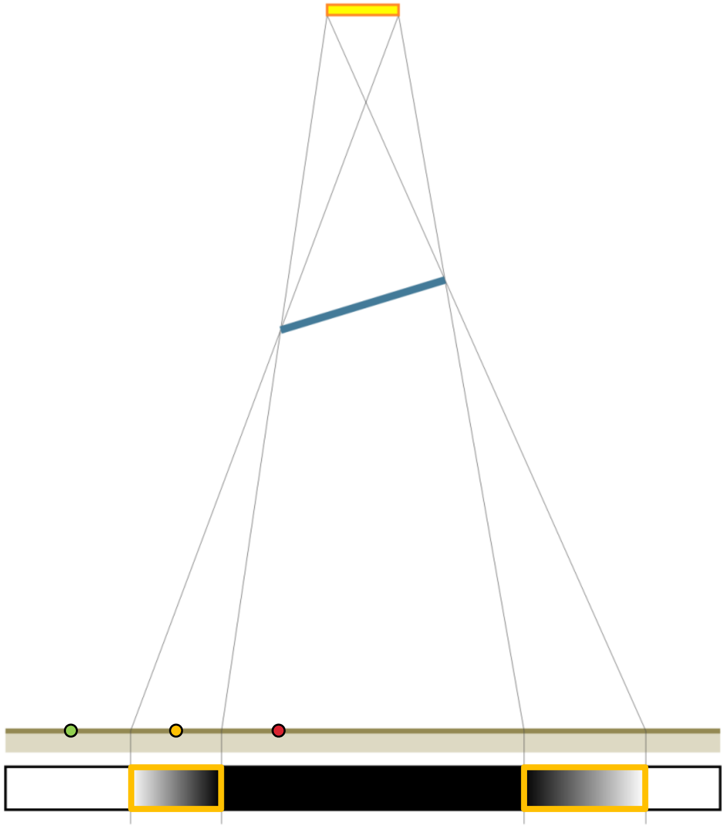
SIGGRAPH2013

Hard Shadows

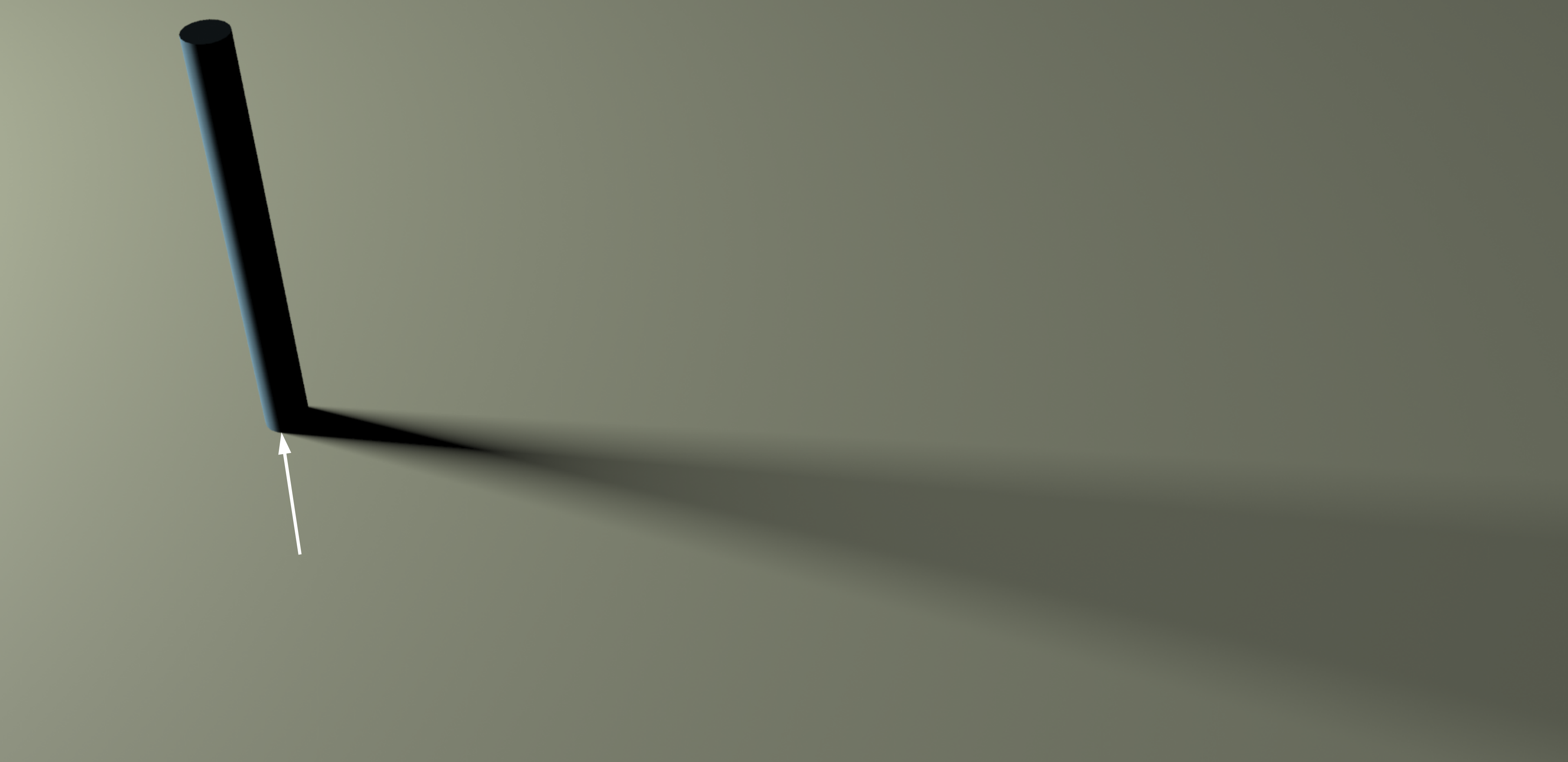




Soft Shadows



Shadow Hardening on Contact

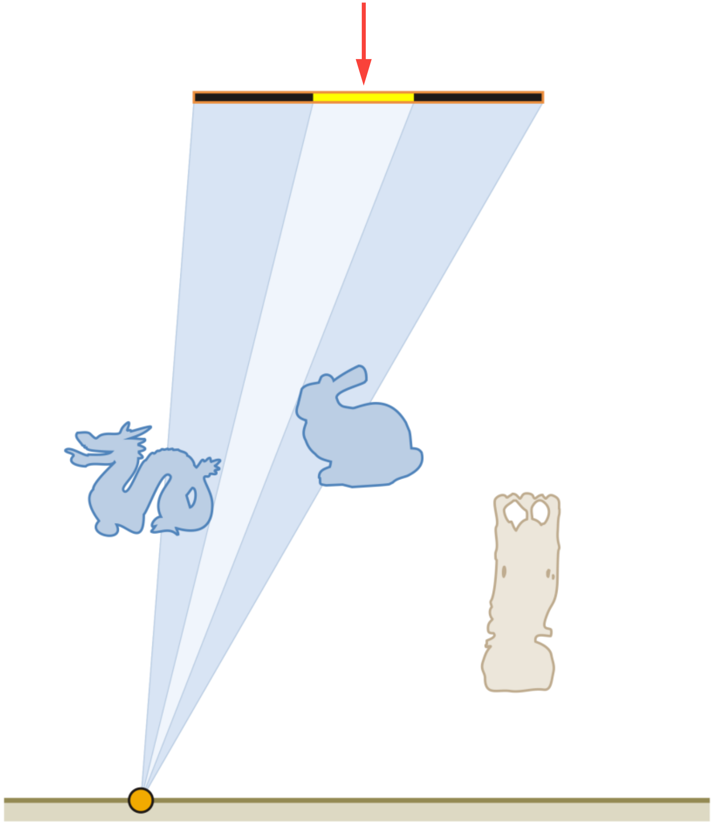


Shadowing = Point–Region Visibility

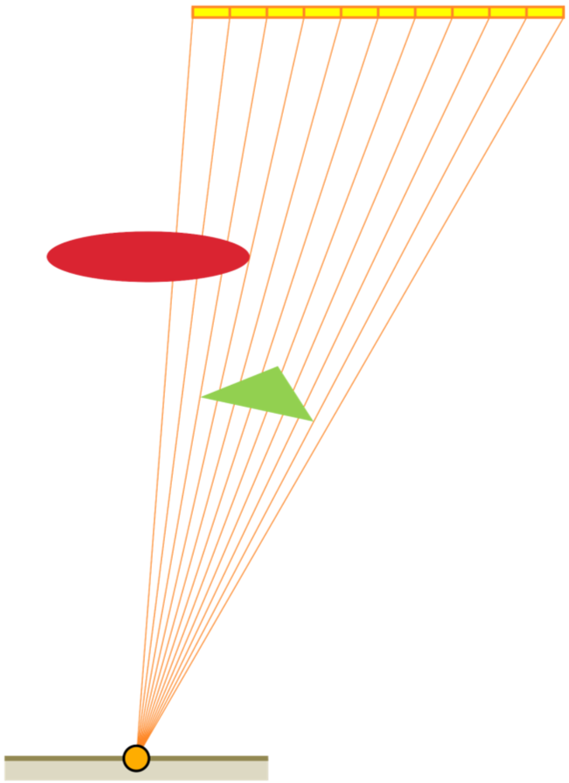
Task:

For each receiver sample (point),

determine visible fraction
of light source (region)



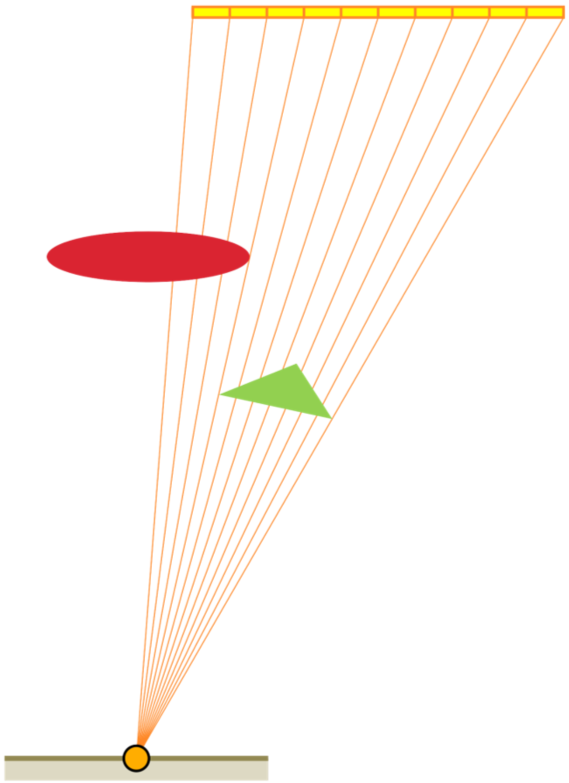
Occluder Fusion



- Occlusion O_1: 30%
- Occlusion O_2: 70%
- Total occlusion: 90%

- $\sum_i O_i$: 100%
- $\max_i O_i$: 70%
- $1 - \prod_i (1 - O_i)$: 79%

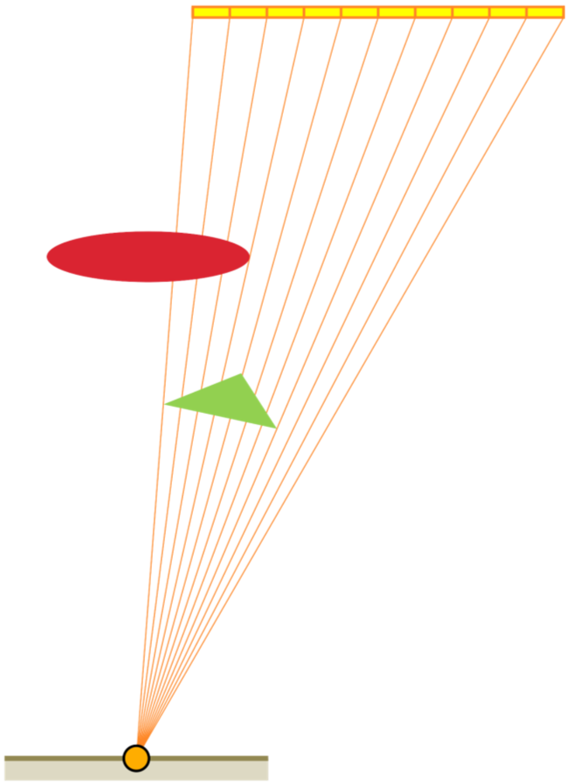
Occluder Fusion



- Occlusion O_1: 30%
- Occlusion O_2: 70%
- Total occlusion: 100%

- $\sum_i O_i$: 100%
- $\max_i O_i$: 70%
- $1 - \prod_i (1 - O_i)$: 79%

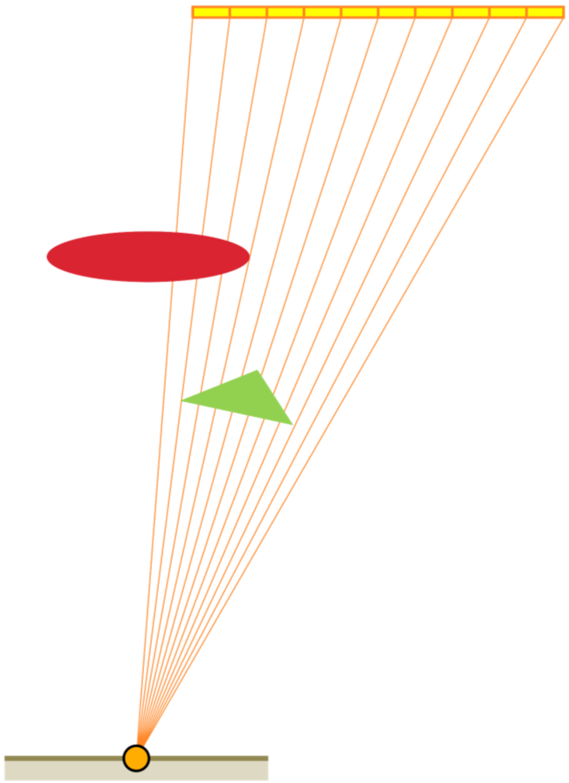
Occluder Fusion



- Occlusion O_1: 30%
- Occlusion O_2: 70%
- Total occlusion: 70%

- $\sum_i O_i$: 100%
- $\max_i O_i$: 70%
- $1 - \prod_i (1 - O_i)$: 79%

Occluder Fusion



- Occlusion O_1: 30%
- Occlusion O_2: 70%
- Total occlusion: 79%

- $\sum_i O_i$: 100%
- $\max_i O_i$: 70%
- $1 - \prod_i (1 - O_i)$: 79%

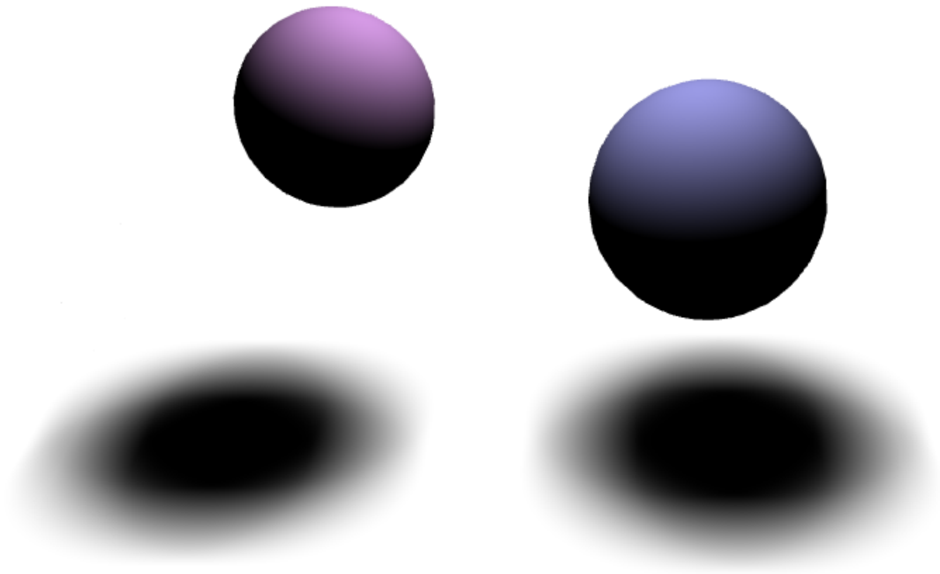
Soft Shadows

Image-based Solutions

Blurring of Hard Shadow Test Results

- Use any of the filtering approaches presented before
- Yields a soft-shadow-like appearance

- But: Ignores varying penumbra width

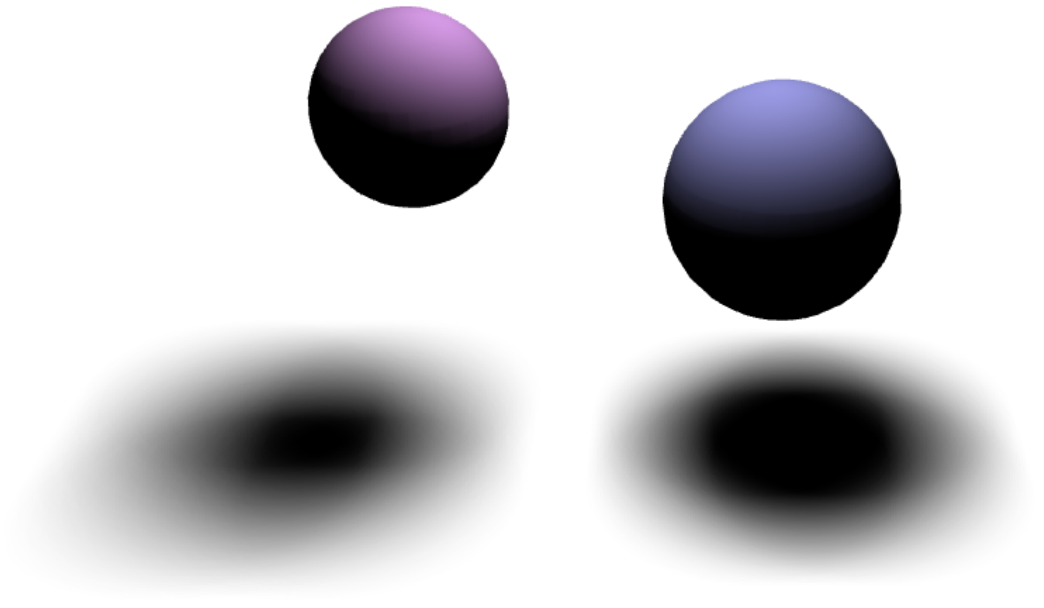


VSM, 512×512, 62×62

Blurring of Hard Shadow Test Results

- Use any of the filtering approaches presented before
- Yields a soft-shadow-like appearance

- But: Ignores varying penumbra width



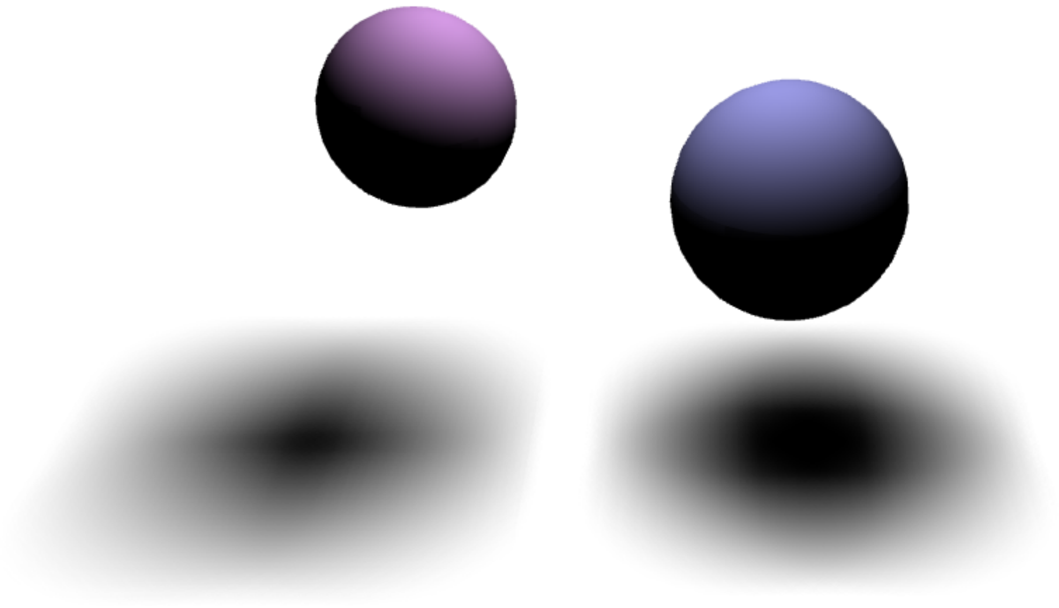
Reference

Blurring of Hard Shadow Test Results

- Idea: Choose blur kernel size adaptively

  But how?

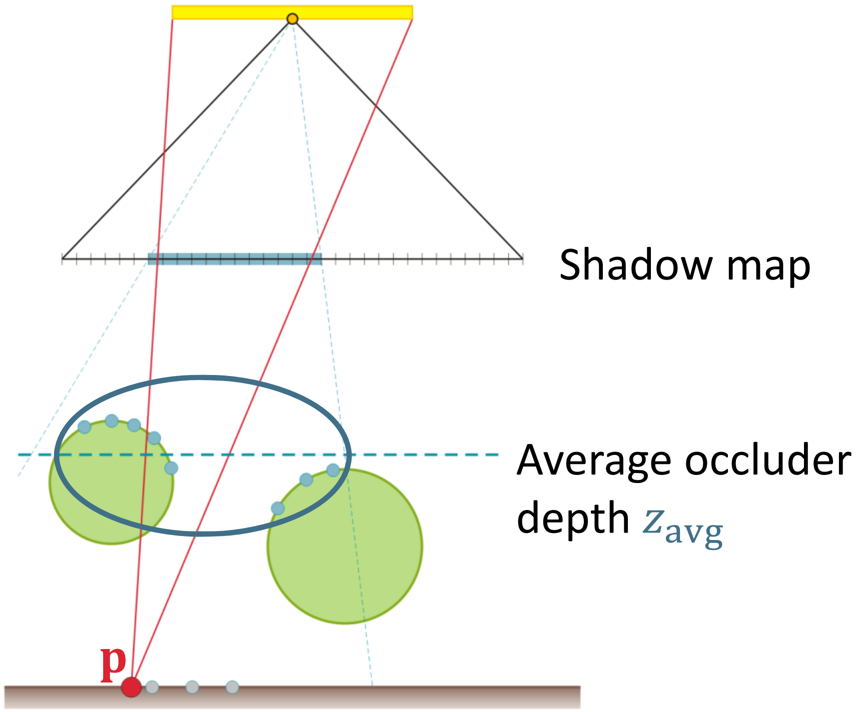
- Percentage-Closer Soft Shadows (PCSS)
  [Fernando, 2005]



PCSS

Percentage-Closer Soft Shadows

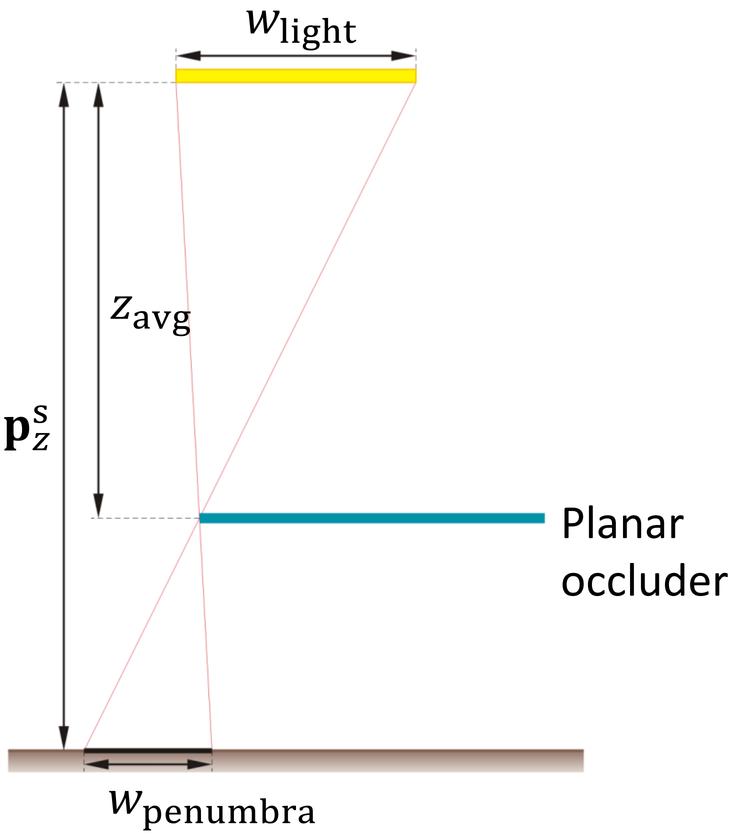
1. Blocker search



Percentage-Closer Soft Shadows

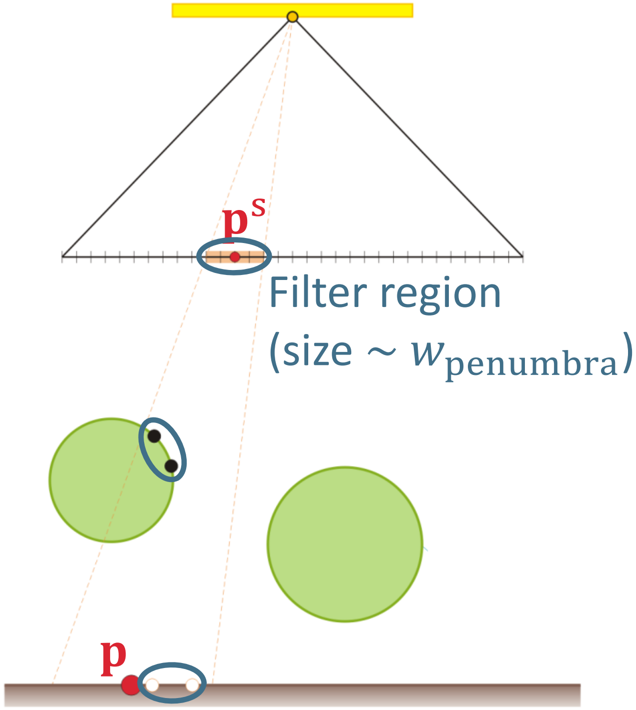
1. Blocker search
2. Penumbra width estimation

$$w_{\text{penumbra}} = \frac{\mathbf{p}_z^s - z_{\text{avg}}}{z_{\text{avg}}} w_{\text{light}}$$



Percentage-Closer Soft Shadows

1. Blocker search
2. Penumbra width estimation
3. Filtering



Percentage-Closer Soft Shadows

1. Blocker search
2. Penumbra width estimation
3. Filtering

Two of these three steps require
many shadow map accesses!

- Acceleration approaches
  - Subsampling
  - Prefiltering

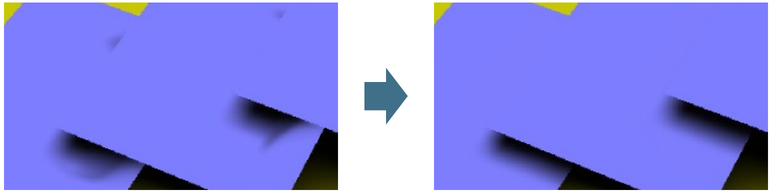
Prefiltering

- Filtering step = Percentage-closer filtering
  - Alternative representation (VSM, CSM, ESM) allows prefiltering
  - Blurring reduces to accessing a mipmap/N-buffers/summed-area table
- Blocker search can also be sped up with prefiltering [Annen et al., 2008]
  - Averaging the depth of shadow map samples closer to the light can be expressed as a convolution
  - Hence approach analogous to CSM is possible ("CSM-Z")
  - Examples: Convolution Soft Shadows,
    Exponential Soft Shadow Mapping [Shen et al., 2013]

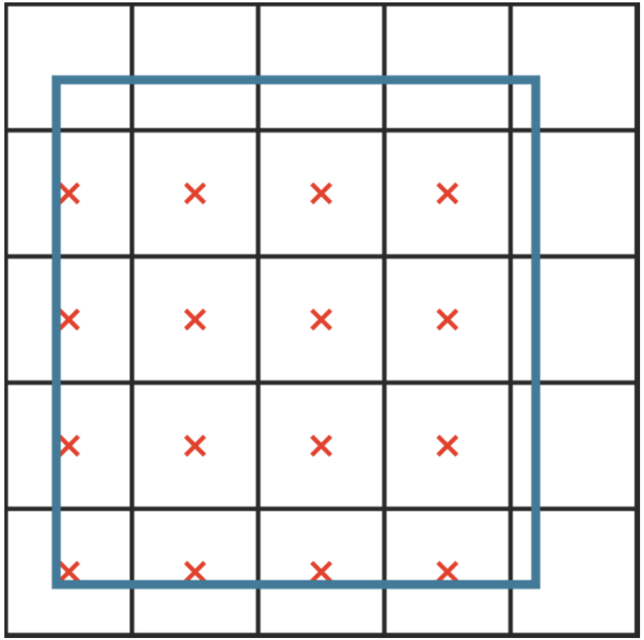
Variance Soft Shadow Mapping

[Yang, Dong et al., 2010]

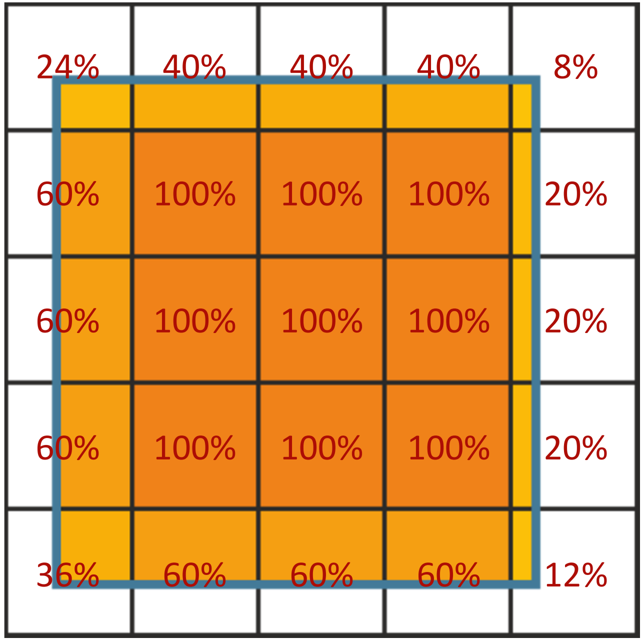
- Estimates average occluder depth directly from prefiltered VSM
  - Expression involves Chebyshev inequality
- Problem: Chebyshev inequality may not be applicable
  - Happens if average depth in query region is larger than receiver depth
  - Conservatively assuming receiver point is fully lit often leads to artifacts
- Improvement: subdivide query region and recurse on subregions
  - If region small and Chebyshev inequality still not applicable: use 2×2 PCF



High-Quality Filtering

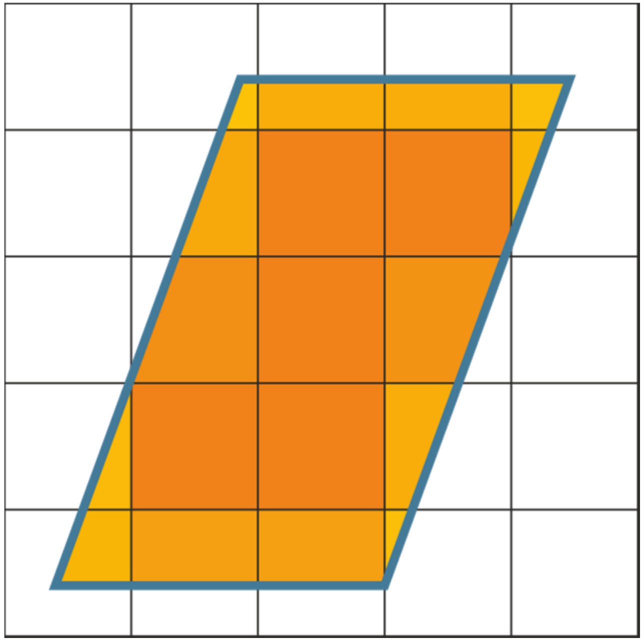


PCF: samples weighted equally

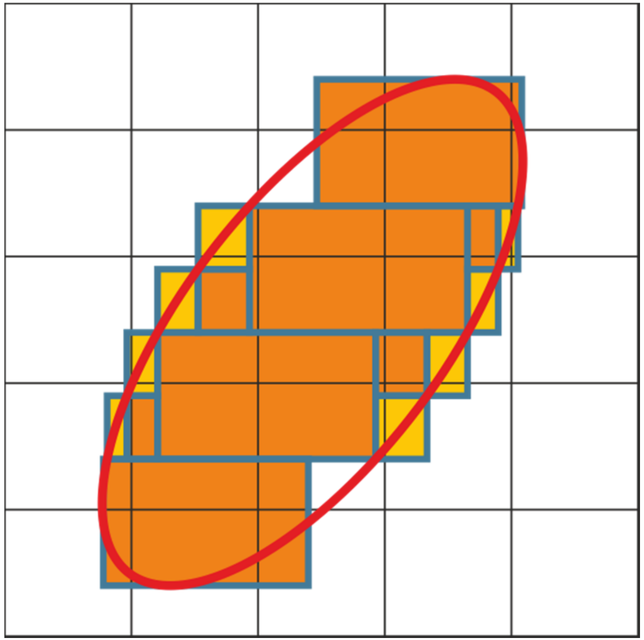


Analytic: actual coverage
[Shen et al., 2011]

Anisotropic Filter Kernels



[Shen et al., 2011]

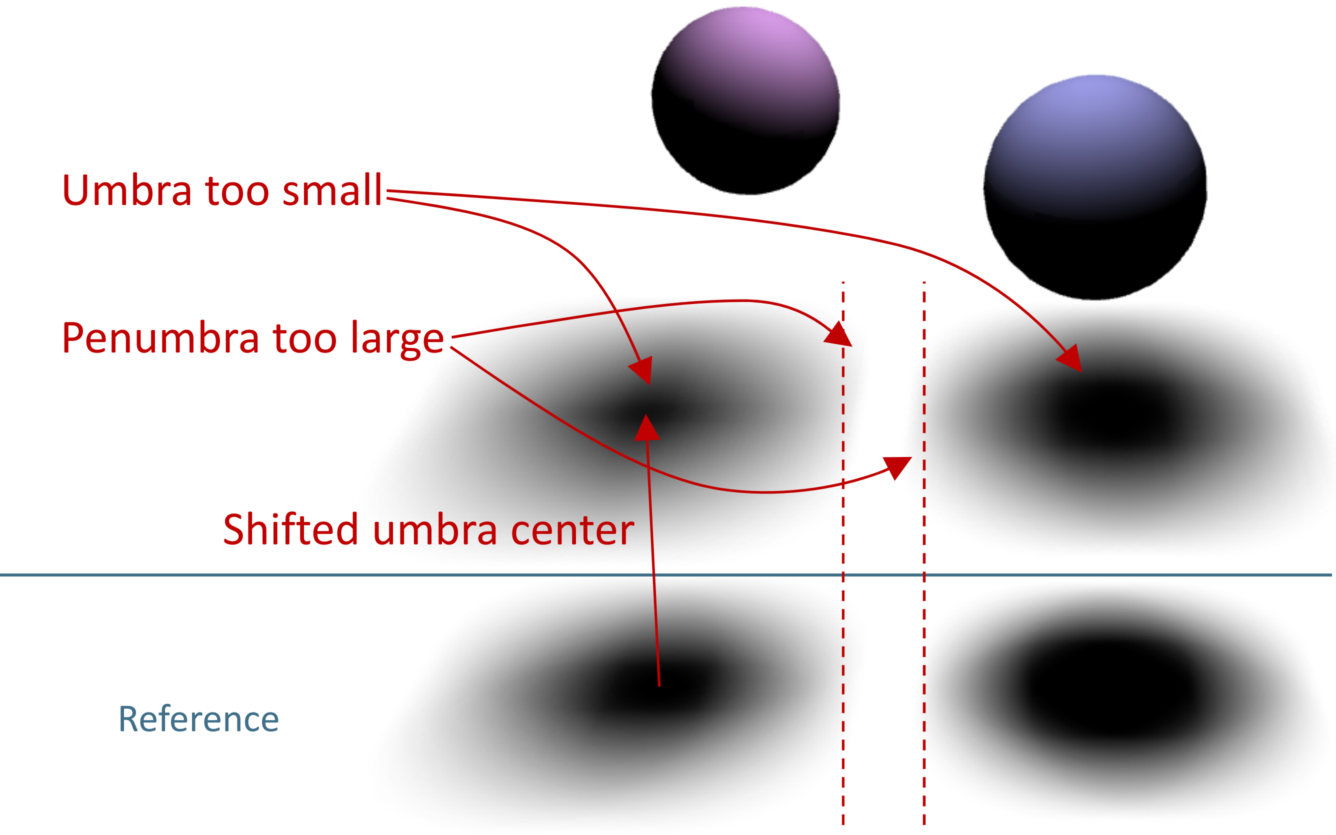


[Shen et al., 2013]

Predicted Virtual Soft Shadow Maps [Shen et al., 2011]

- Adaptive shadow-map partitioning
  - Guided by perceptual resolution prediction metric
  - Accounts for screen-space frequency of penumbrae

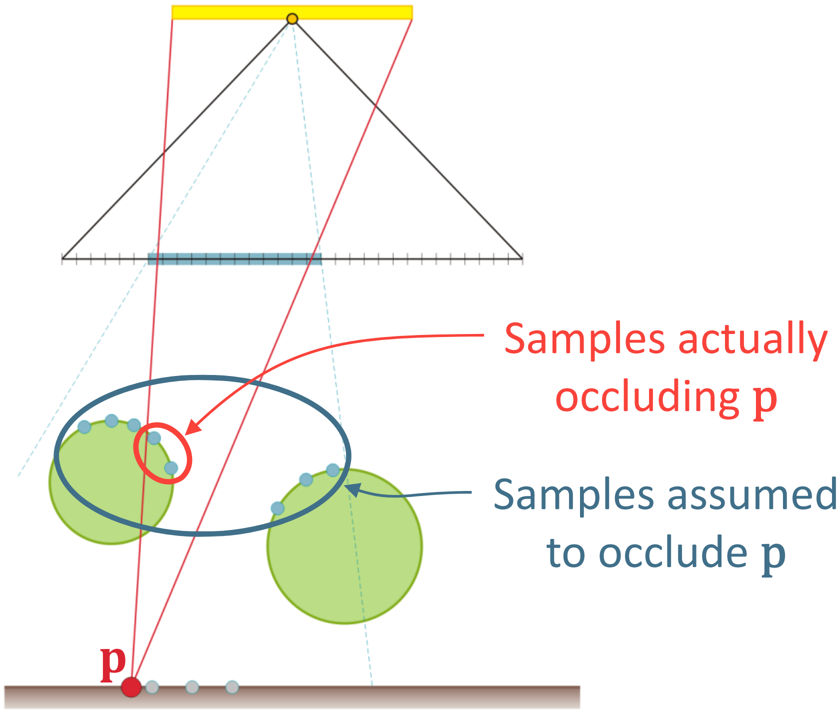
Percentage-Closer Soft Shadows



Main sources of incorrectness

- Single planar occluder assumption
- Classification as light blocking solely based on depth test

Percentage-Closer Soft Shadows



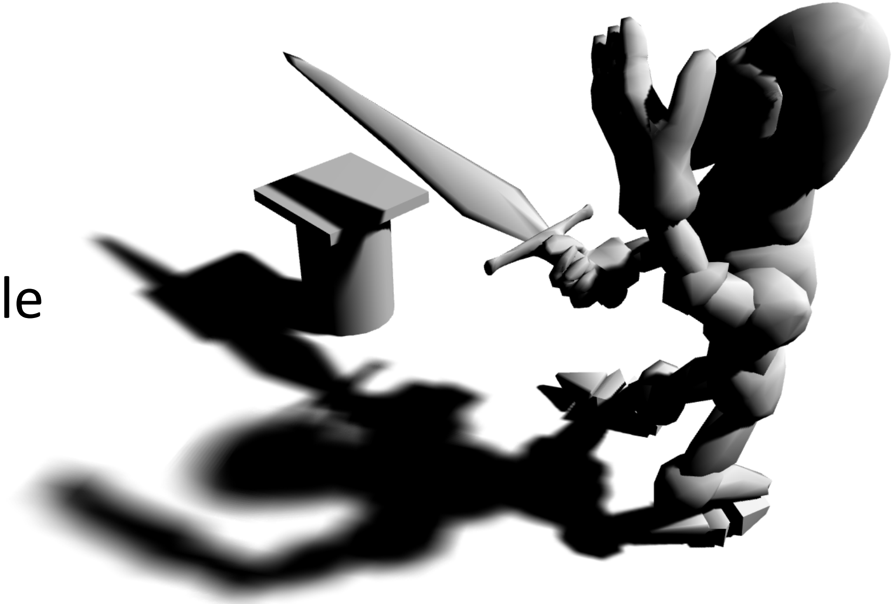
Main sources of incorrectness

- Single planar occluder assumption
- Classification as light blocking solely based on depth test

Percentage-Closer Soft Shadows

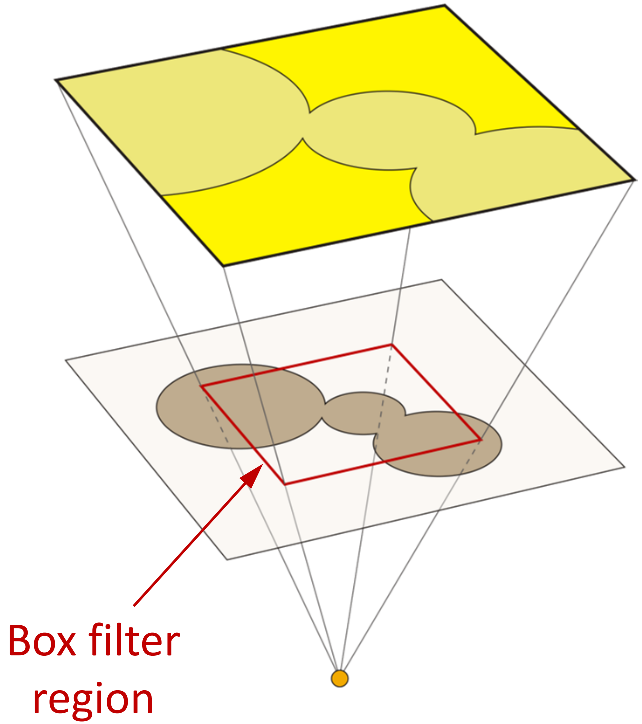
- \+ Simple and reasonably fast
- \+ Often visually pleasing results
  (at least for smaller light sources)

- – Not really physically plausible
- – Only accounts for occluders visible from light source's center

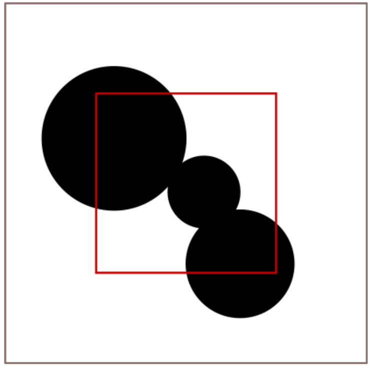


Visibility Computation = Filtering [Soler and Sillion, 1998]

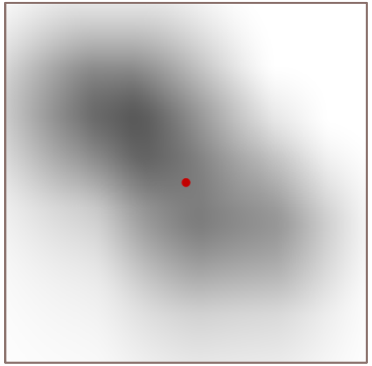
- Rectangular light source
- Planar occluder parallel to light
  - Represented by blocker image
- Visibility factor is obtained via box filtering the blocker image
  - Filter size equals appropriately scaled light size



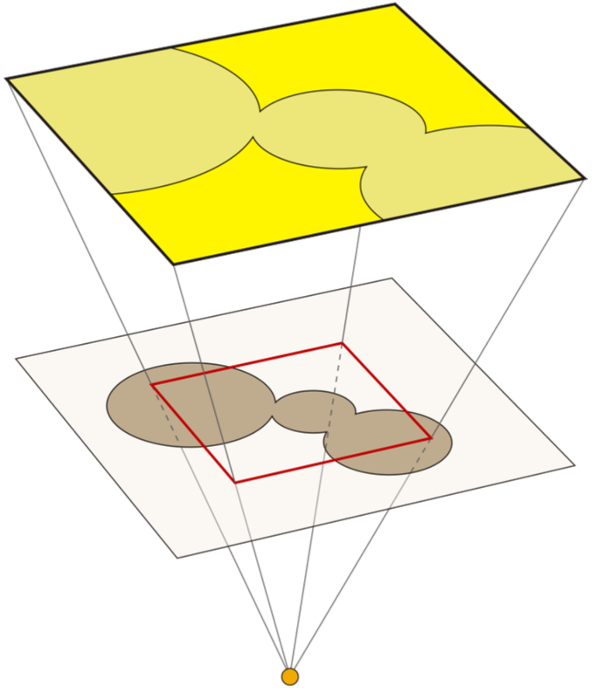
Visibility Computation = Filtering



Blocker image



Box-filtered blocker image



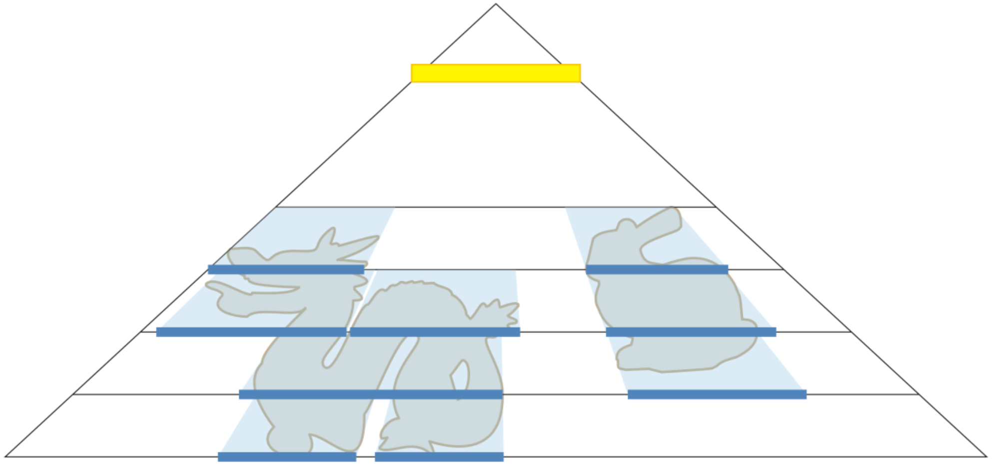
Advantages

- Enables prefiltering
- Query becomes constant-time (independent of light size)

Occlusion Textures

[Eisemann and Décoret, 2006]

- Decompose scene into multiple planar layers
  - Slice scene parallel to light source
  - Project geometry within slice onto slice's bottom plane



Occlusion Textures

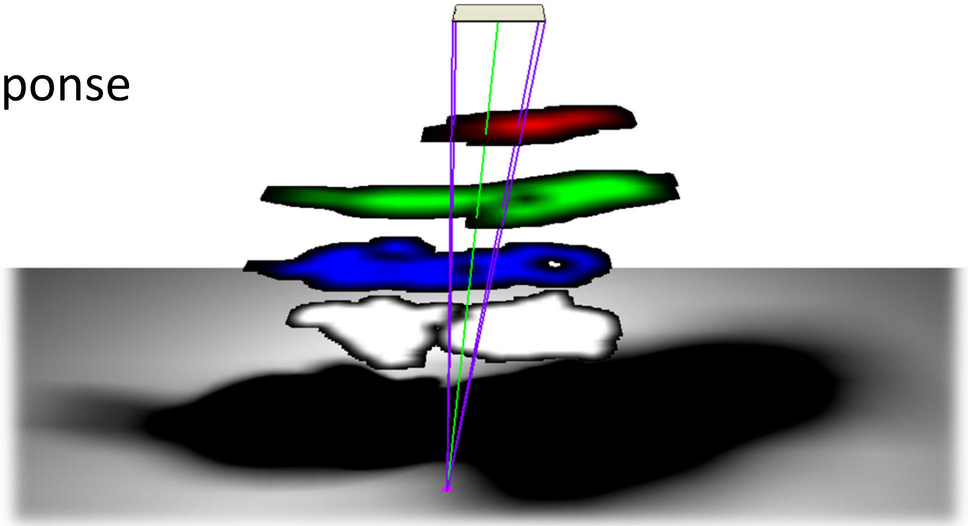
[Eisemann and Décoret, 2006]

- Decompose scene into multiple planar layers
  - Slice scene parallel to light source
  - Project geometry within slice onto slice's bottom plane

- Covered parts of each slice are encoded in a binary occlusion texture (= blocker image)

Occlusion Textures

- Prefilter occlusion textures
- At each view sample
  - For each blocking slice,
    lookup appropriately filtered response
    in prefiltered occlusion texture
  - Accumulate per-slice
    shadow contributions



Occlusion Textures

- + Plausible soft shadows at high frame rates
- + Performance independent of light size

- Mainly suited for compact indoor environments

- – Heuristic occluder fusion
- – Discretizes occluders into small number of perforated planes

Occluder Backprojection

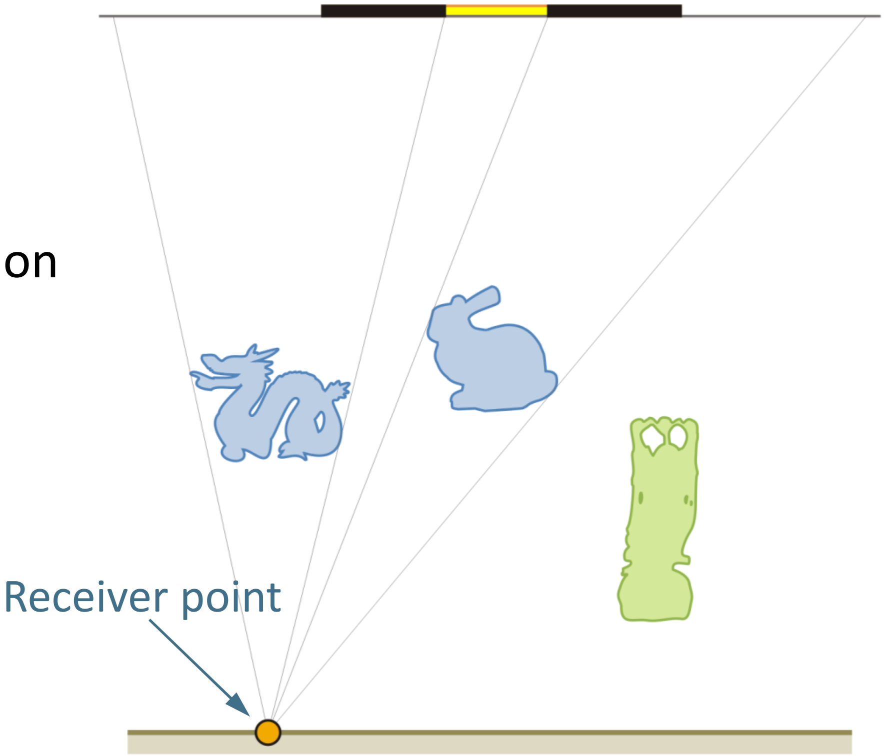
- For each (relevant) occluder
  - Project it onto light source
  - Determine covered light area
  - Aggregate this occlusion information
- Gathering

  For all receiver points

      For all potential occluders
- Scattering

  For all occluders

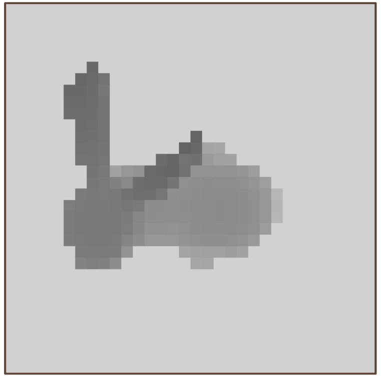
      For all affected receiver points

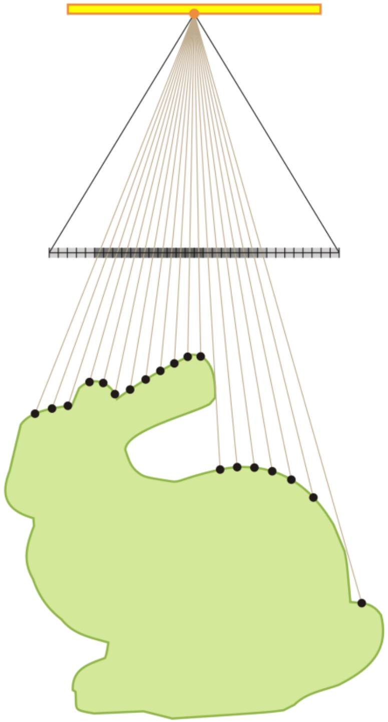


Soft Shadow Mapping

[Atty et al., 2006; Guennebaud et al., 2006]

- Approximate (subset of) occluder geometry
- Generate shadow map
  (from light's center)

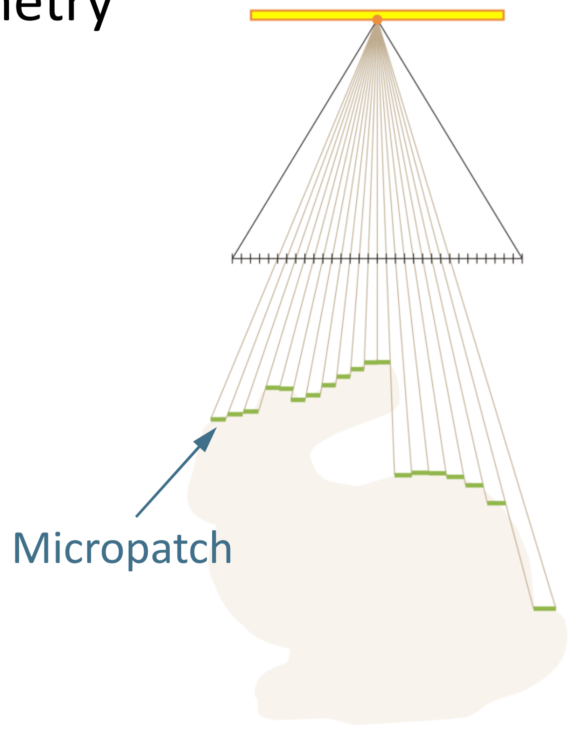




Soft Shadow Mapping

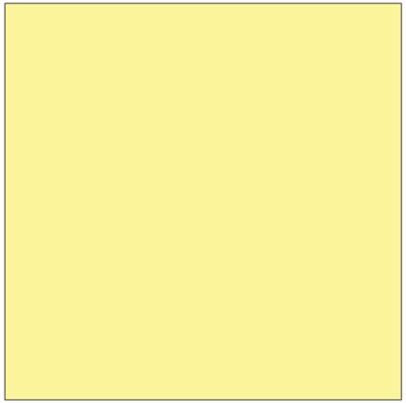
[Atty et al., 2006; Guennebaud et al., 2006]

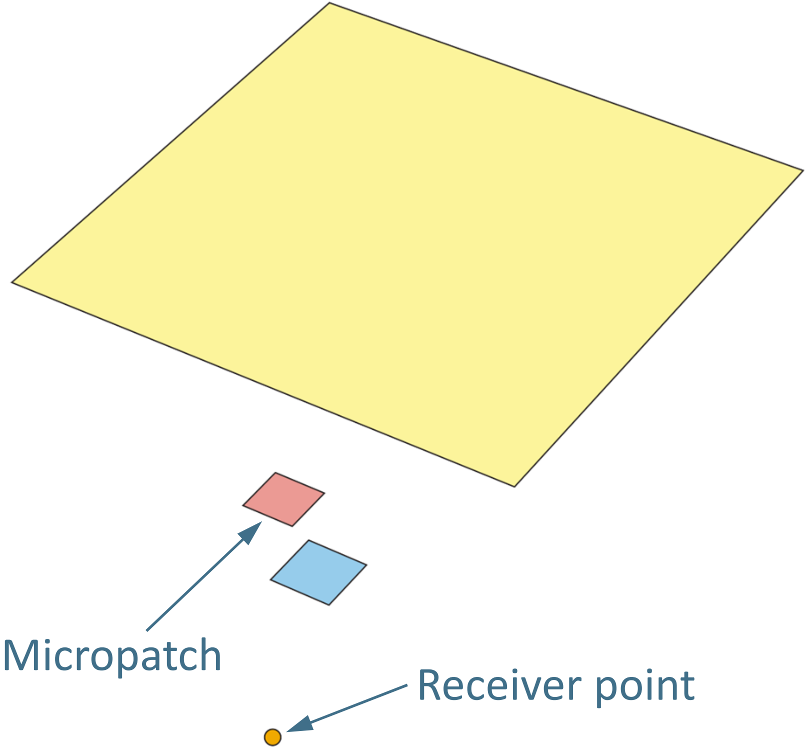
- Approximate (subset of) occluder geometry
- Generate shadow map
  (from light's center)
- Derive occluder approximation
  by unprojecting texels into world space
  ➡ Micropatches



Soft Shadow Mapping

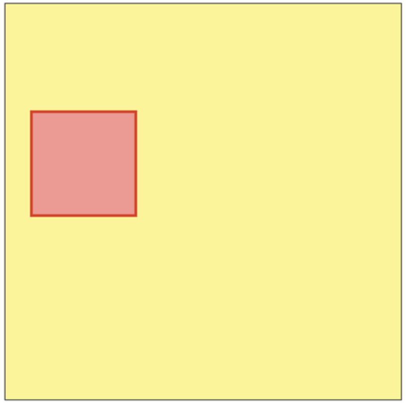
- Backproject micropatches onto light source to determine visibility
- Simple approach:
  Sum up projections' covered areas
  - Ignores overlaps

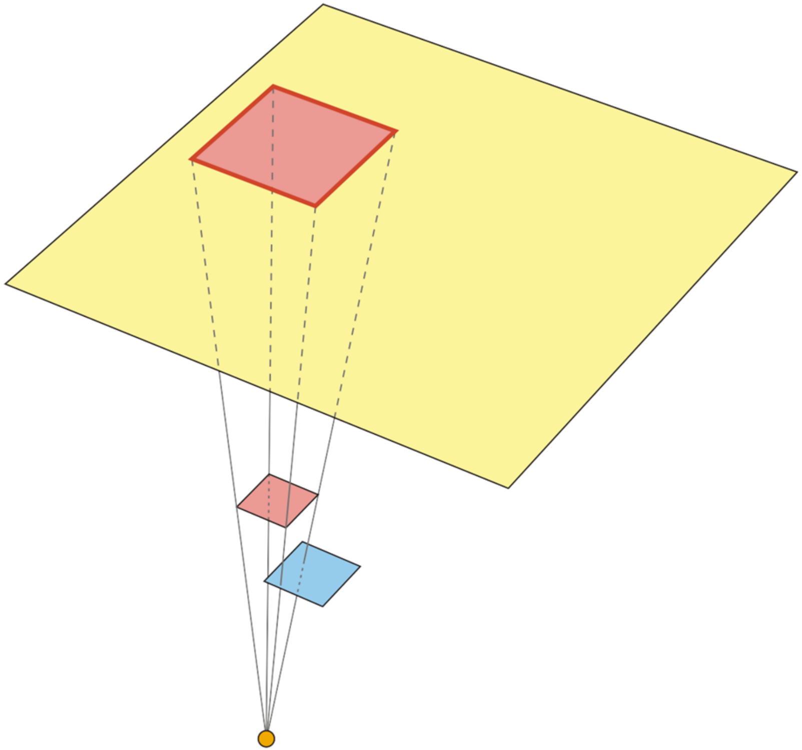




Soft Shadow Mapping

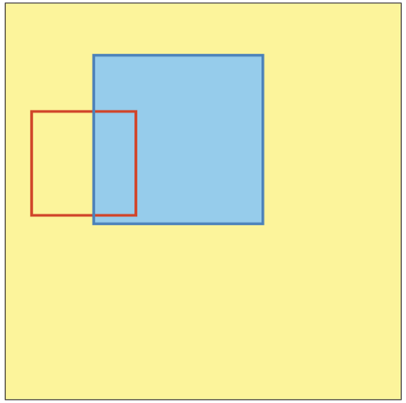
- Backproject micropatches onto light source to determine visibility
- Simple approach:
  Sum up projections’ covered areas
  - Ignores overlaps

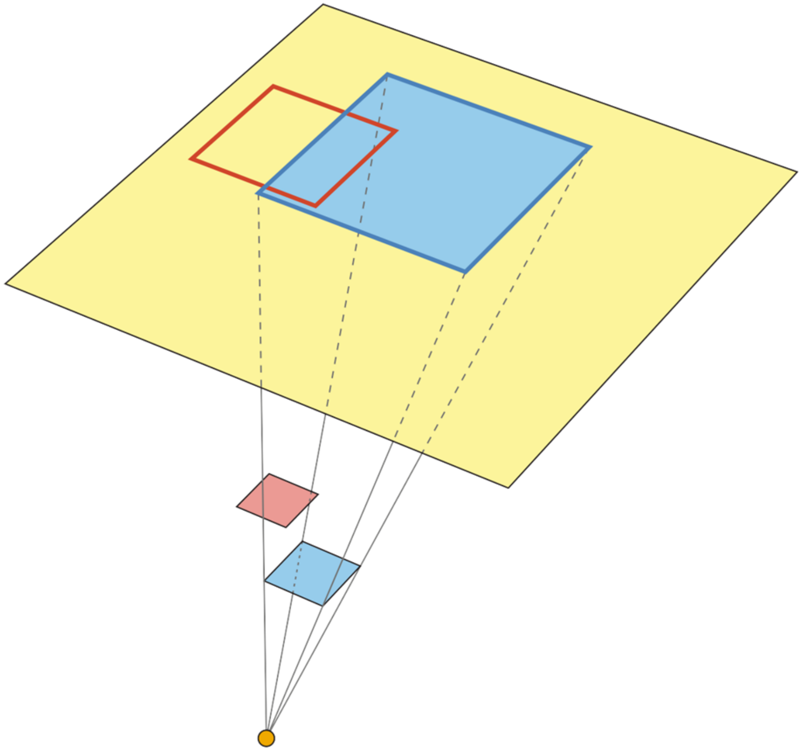




Soft Shadow Mapping

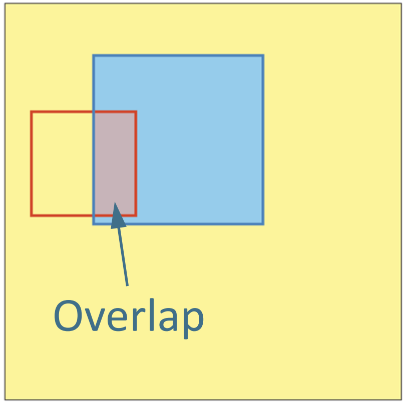
- Backproject micropatches onto light source to determine visibility
- Simple approach:
  Sum up projections' covered areas
  - Ignores overlaps

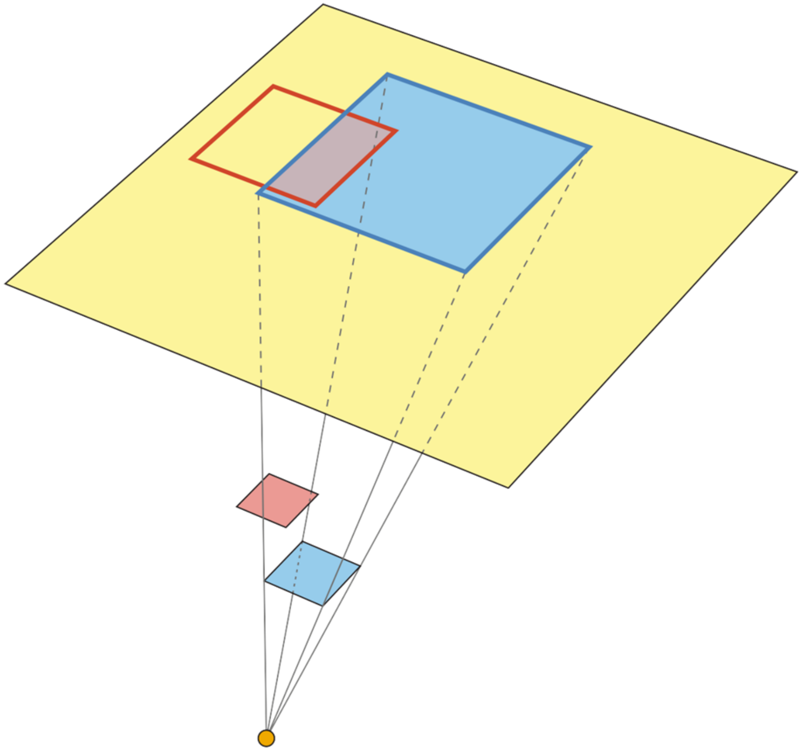




Soft Shadow Mapping

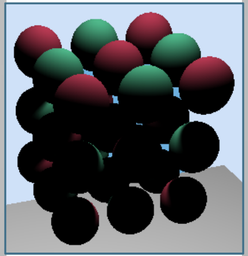
- Backproject micropatches onto light source to determine visibility
- Simple approach:
  Sum up projections' covered areas
  - Ignores overlaps





Overlapping Artifacts

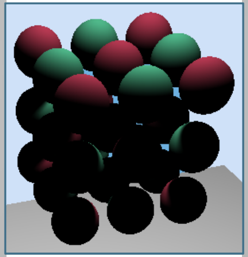
Reference

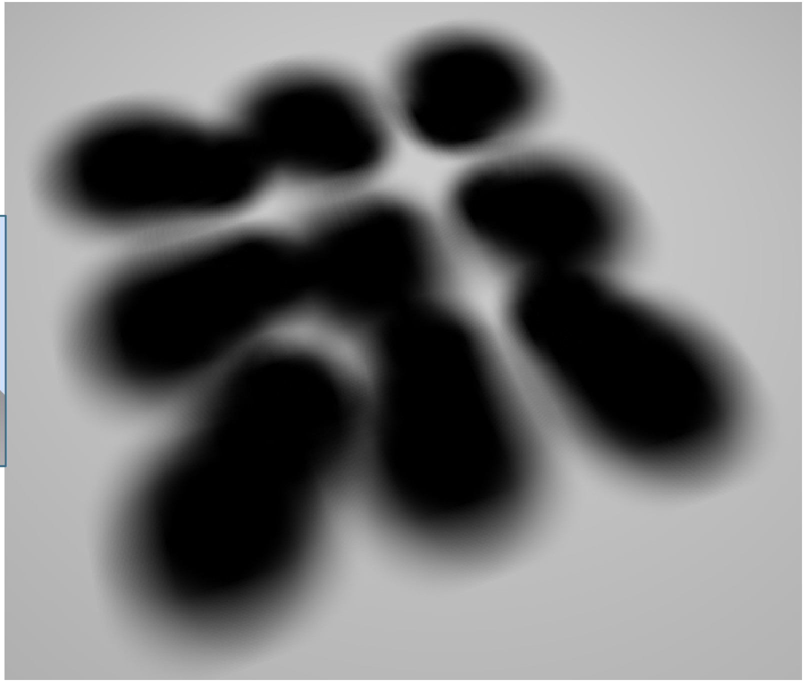


Area accumulation
(without accounting for overlaps)

Overlapping Artifacts

Reference



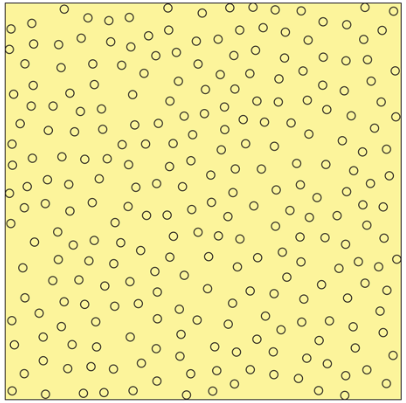


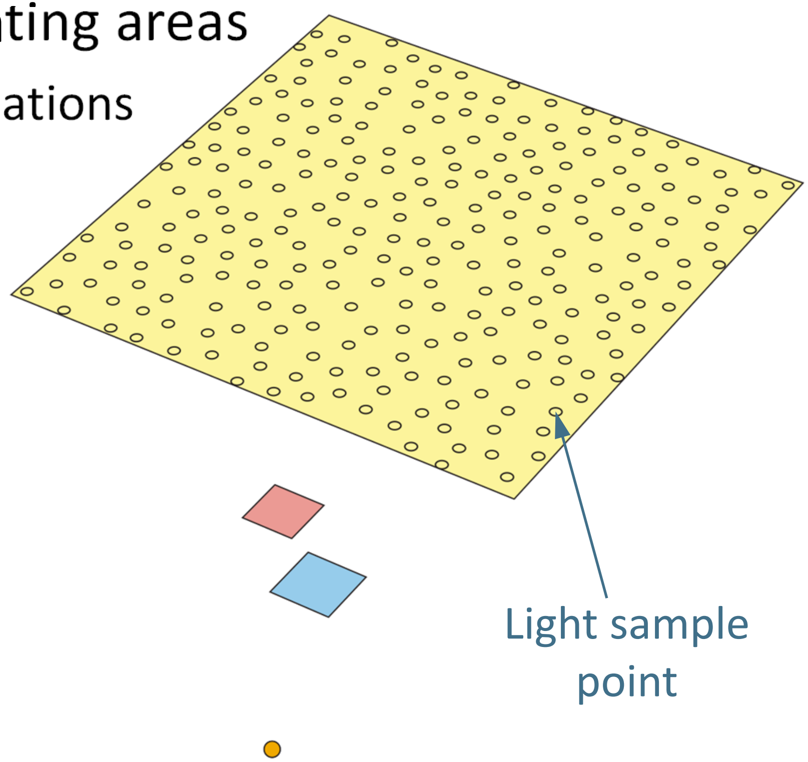
Occlusion bitmasks
(with correct overlap handling)

Occlusion Bitmasks

[Schwarz & Stamminger, 2007]

- Sample visibility instead of accumulating areas
  - Set of binary point-to-point visibility relations
  - Bit field is employed to track visibilities of sample points on light source

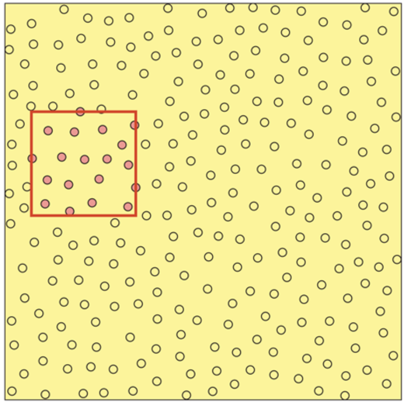


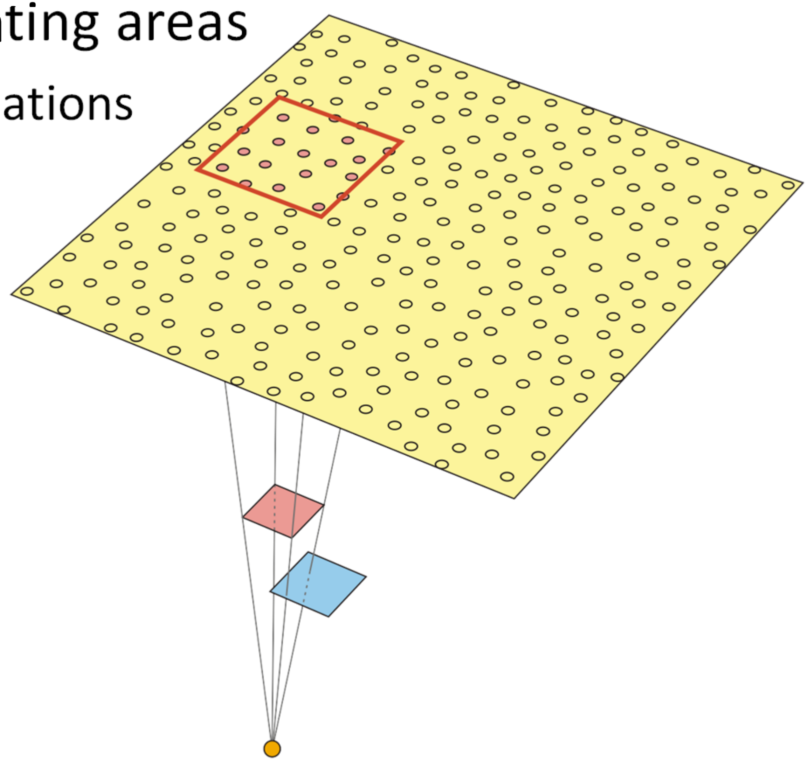


Occlusion Bitmasks

[Schwarz & Stamminger, 2007]

- Sample visibility instead of accumulating areas
  - Set of binary point-to-point visibility relations
  - Bit field is employed to track visibilities of sample points on light source

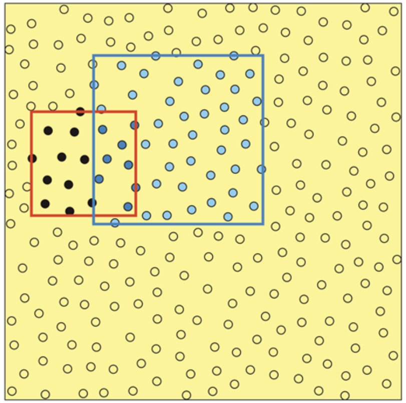


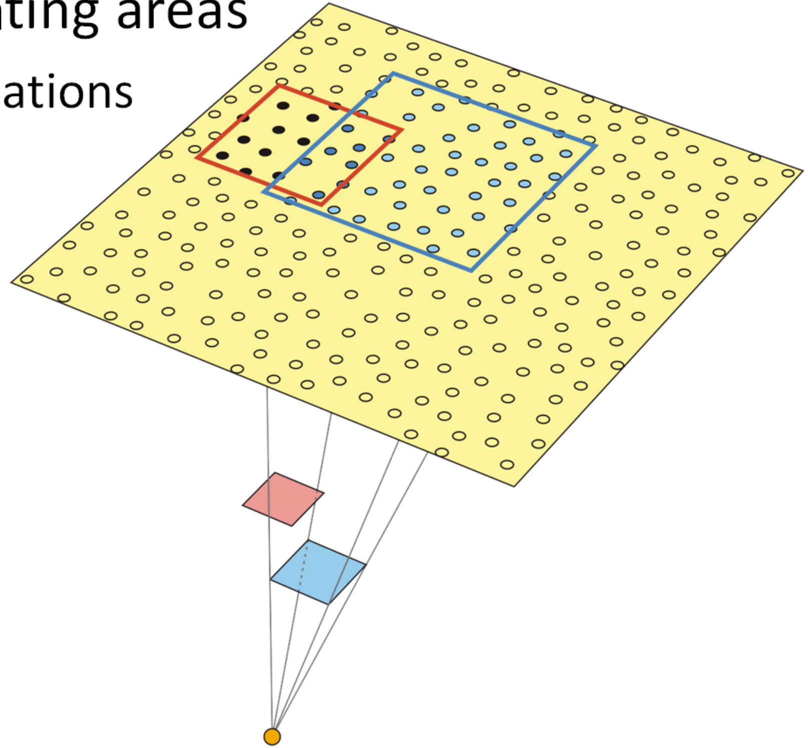


Occlusion Bitmasks

[Schwarz & Stamminger, 2007]

- Sample visibility instead of accumulating areas
  - Set of binary point-to-point visibility relations
  - Bit field is employed to track visibilities of sample points on light source

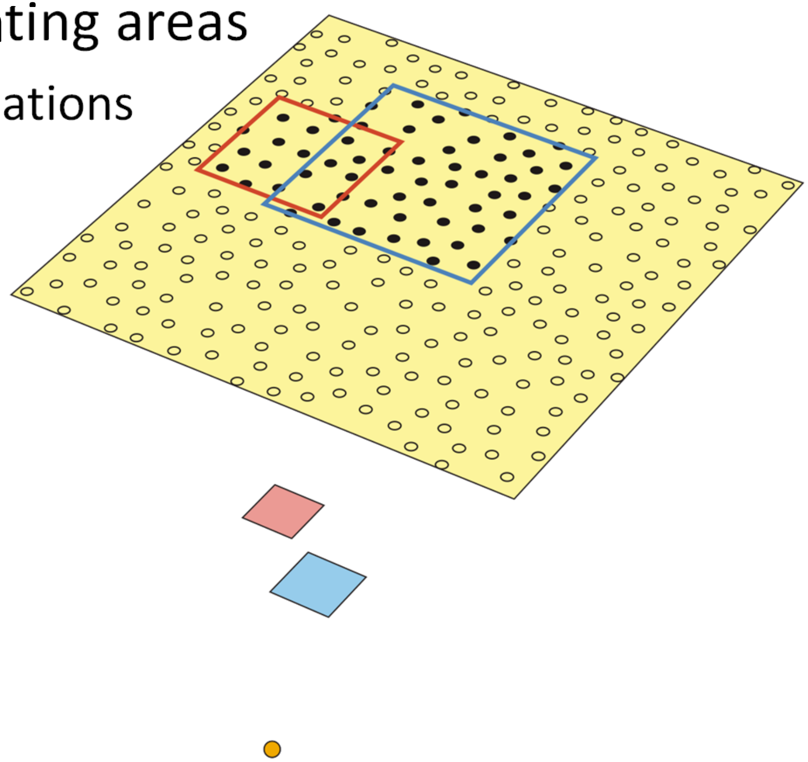


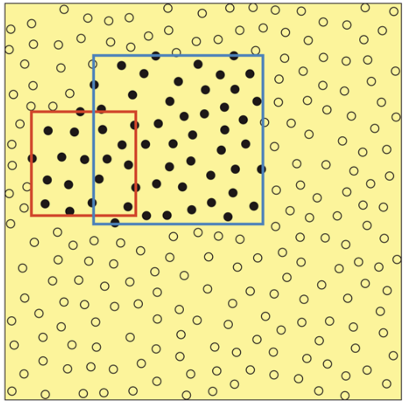


Occlusion Bitmasks

[Schwarz & Stamminger, 2007]

- Sample visibility instead of accumulating areas
  - Set of binary point-to-point visibility relations
  - Bit field is employed to track visibilities of sample points on light source

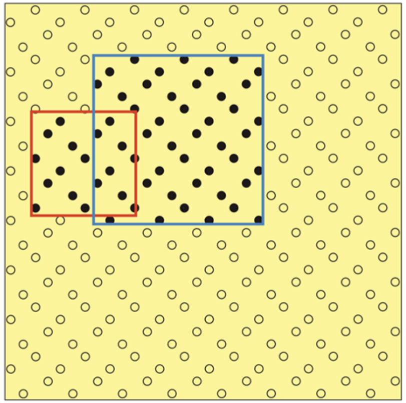




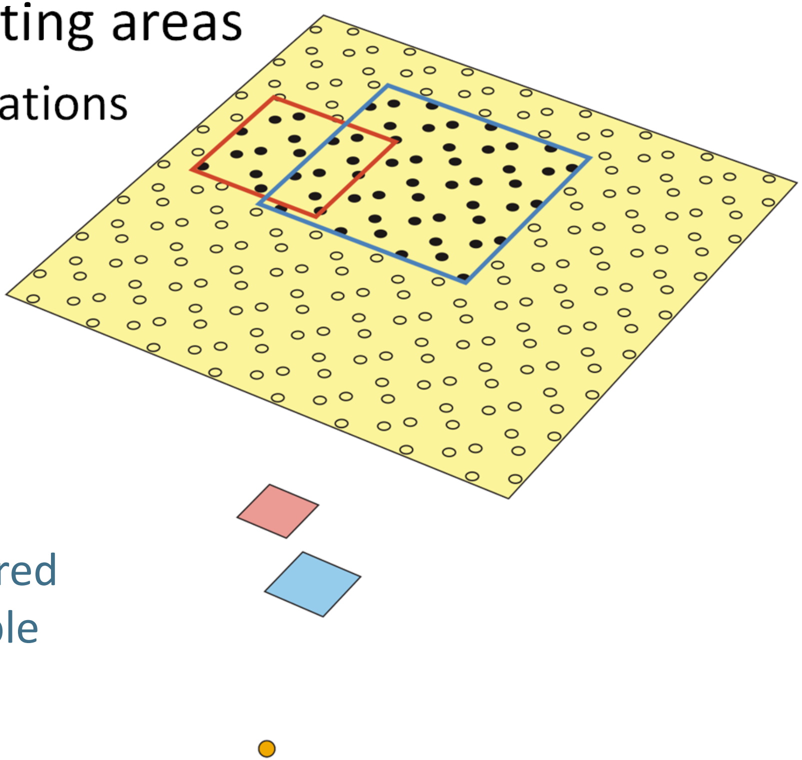
Occlusion Bitmasks

[Schwarz & Stamminger, 2007]

- Sample visibility instead of accumulating areas
  - Set of binary point-to-point visibility relations
  - Bit field is employed to track visibilities of sample points on light source



16×16 jittered light sample points



Many Options

- Visibility determination methods
  - Area accumulation
  - Occlusion bitmasks

Many Options

- Visibility determination methods
- Occluder approximations
  - Micropatches
  - Microquads (and microtris)
  - Occluder contours

Many Options

- Visibility determination methods
- Occluder approximations
- Acceleration by adapting accuracy
  - Micro-occluder subsampling
  - Coarser occluder approximations
  - Subsampling in screen space

Many Options

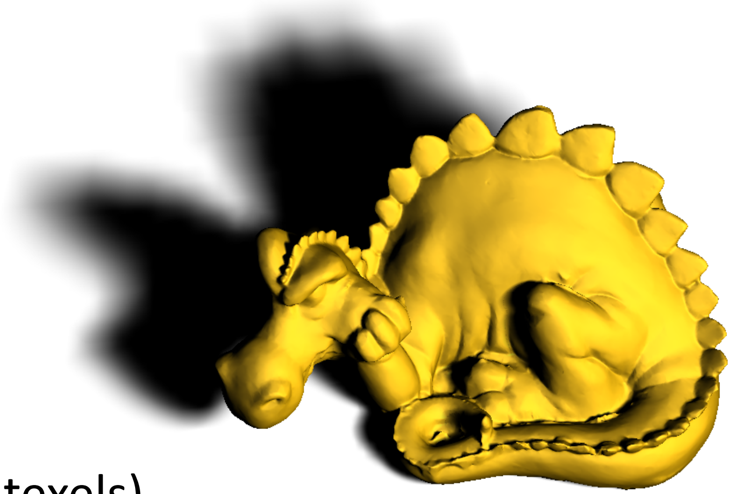
- Visibility determination methods
- Occluder approximations
- Acceleration by adapting accuracy
- Acceleration via multi-scale (min/max) representations
  - Search area pruning
  - Direct identification of umbra and fully-lit receiver points
  - Hierarchical occluder construction

Many Options

- Visibility determination methods
- Occluder approximations
- Acceleration by adapting accuracy
- Acceleration via multi-scale (min/max) representations

- Practical example: variant by Geomerics [Martin, 2012]
  - Targets large area lights
  - CUDA-based implementation

Soft Shadow Mapping



- \+ Physically plausible
- \+ Rather high quality
  at real-time frame rates possible
- Performance strongly dependent on
  - Number of pixels requiring backprojection
  - Search area size (number of backprojected texels)
- – Uses only approximation of subset of occluders
  - Typically those visible from the light source’s center