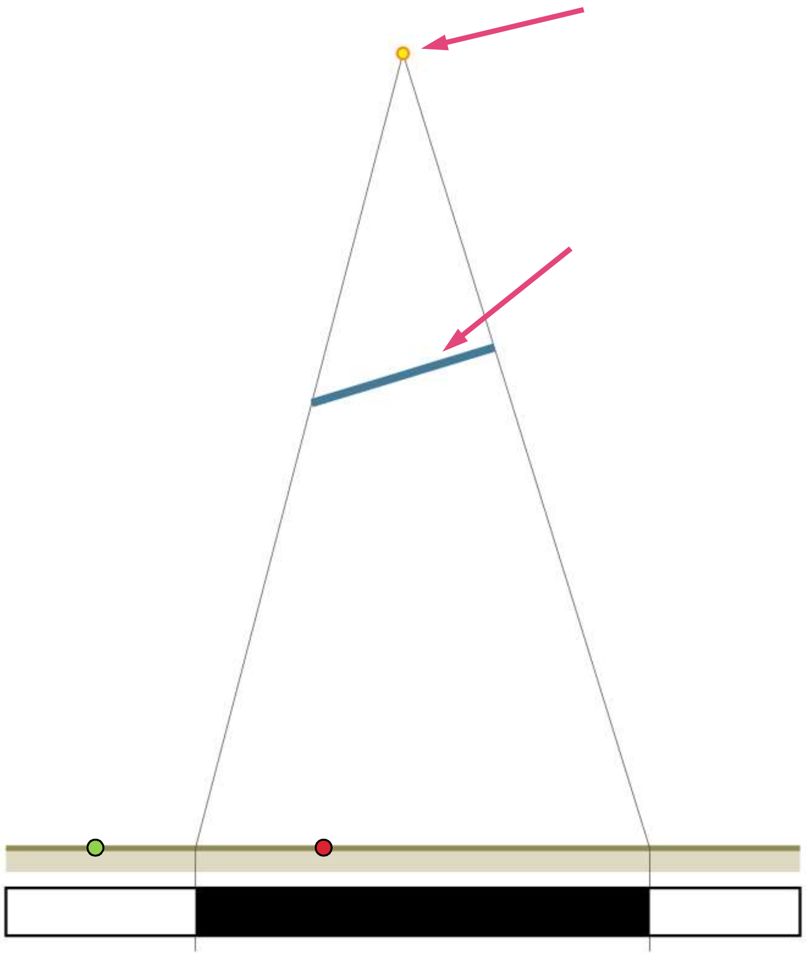
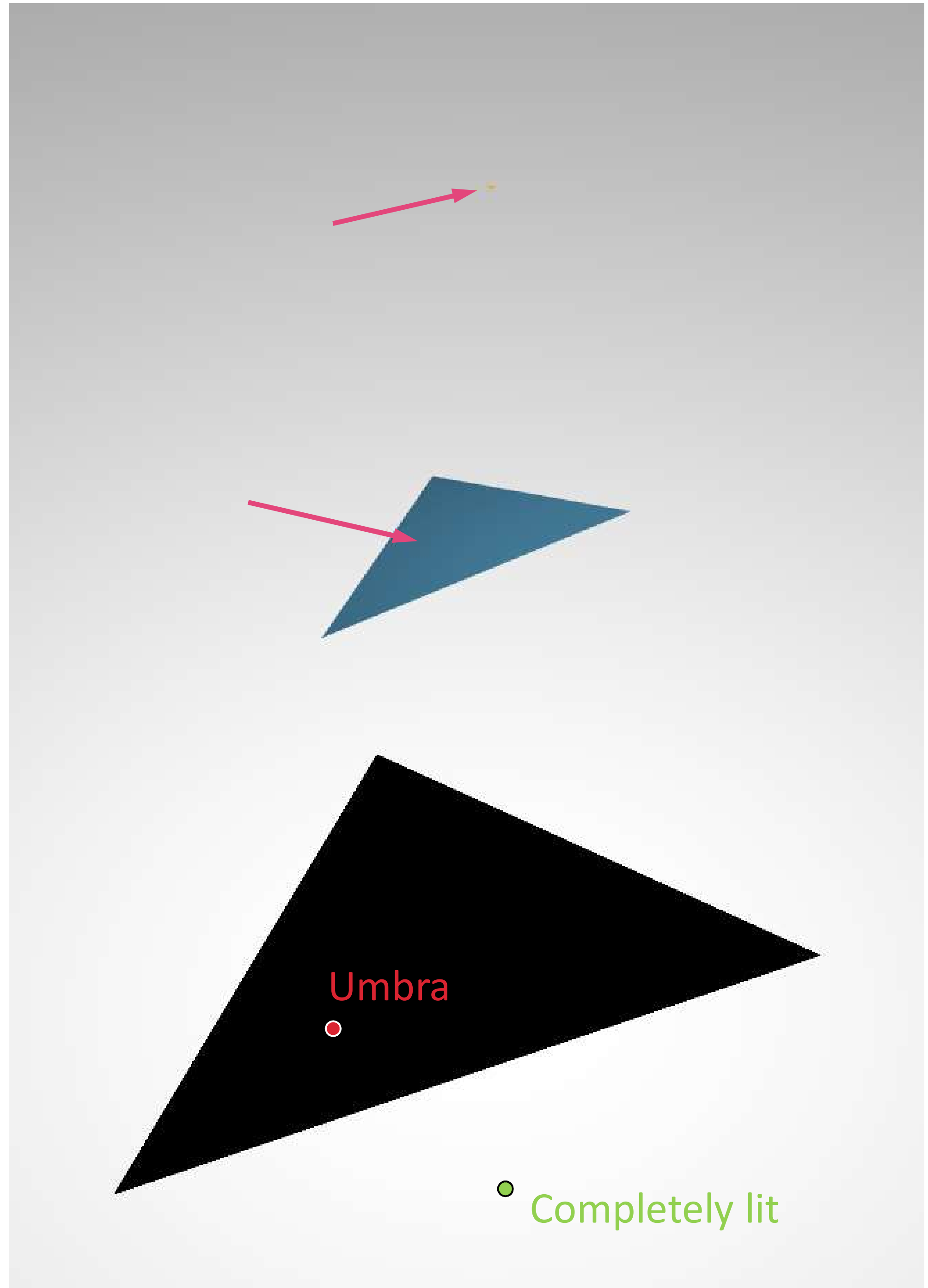
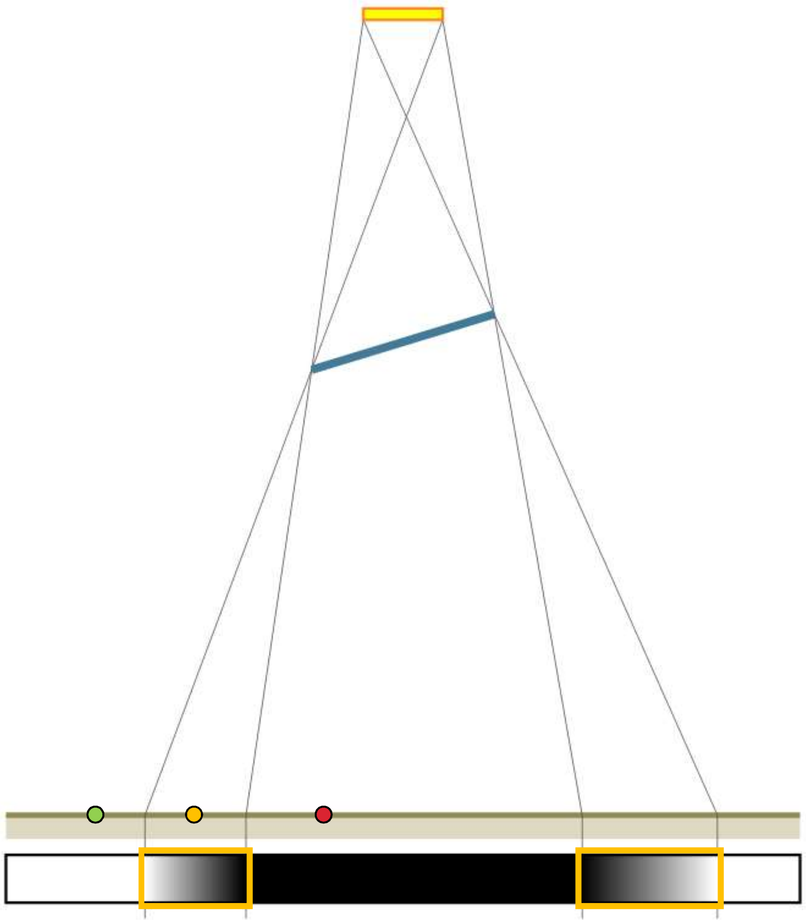
# Soft Shadows

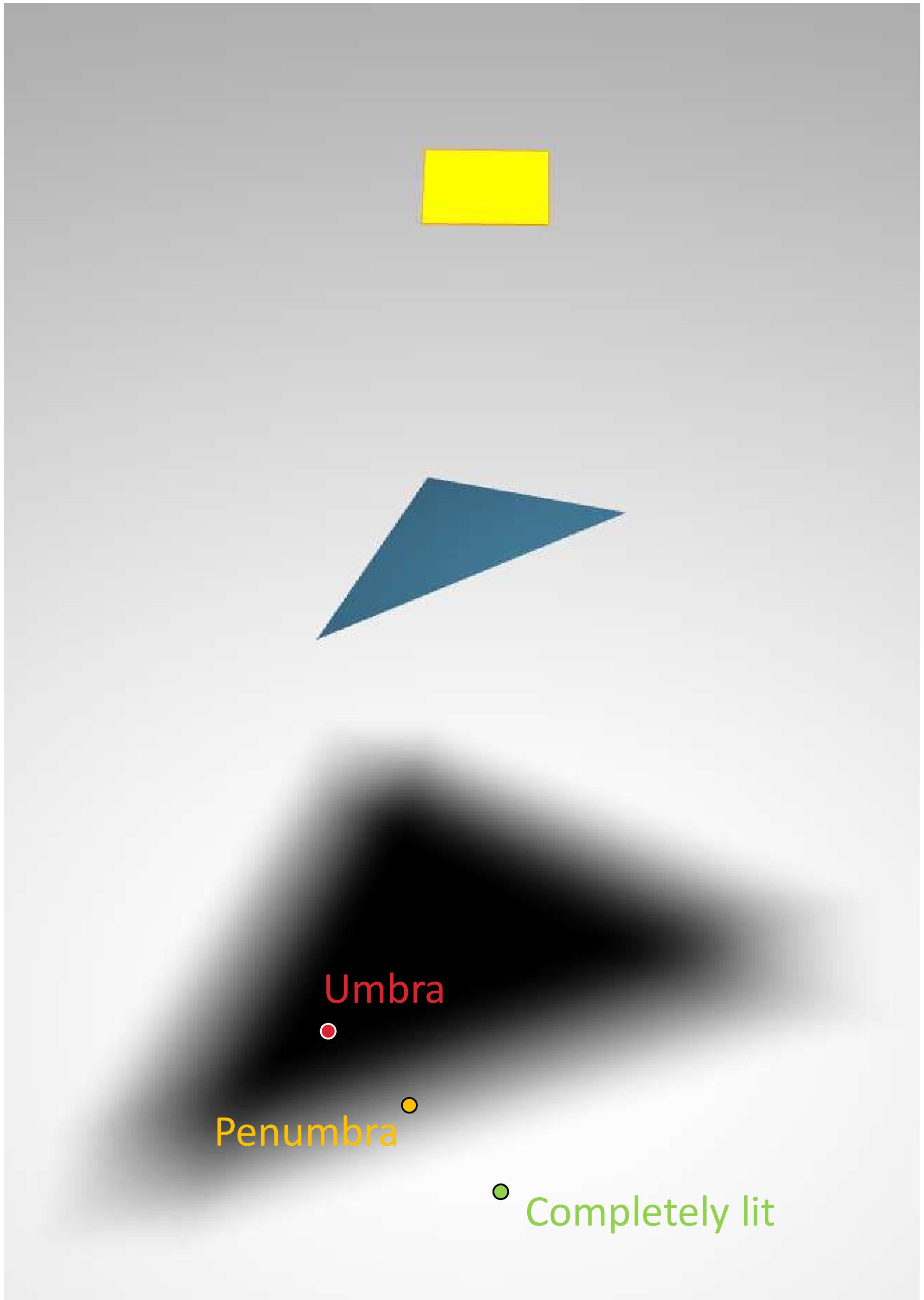
# Hard Shadows



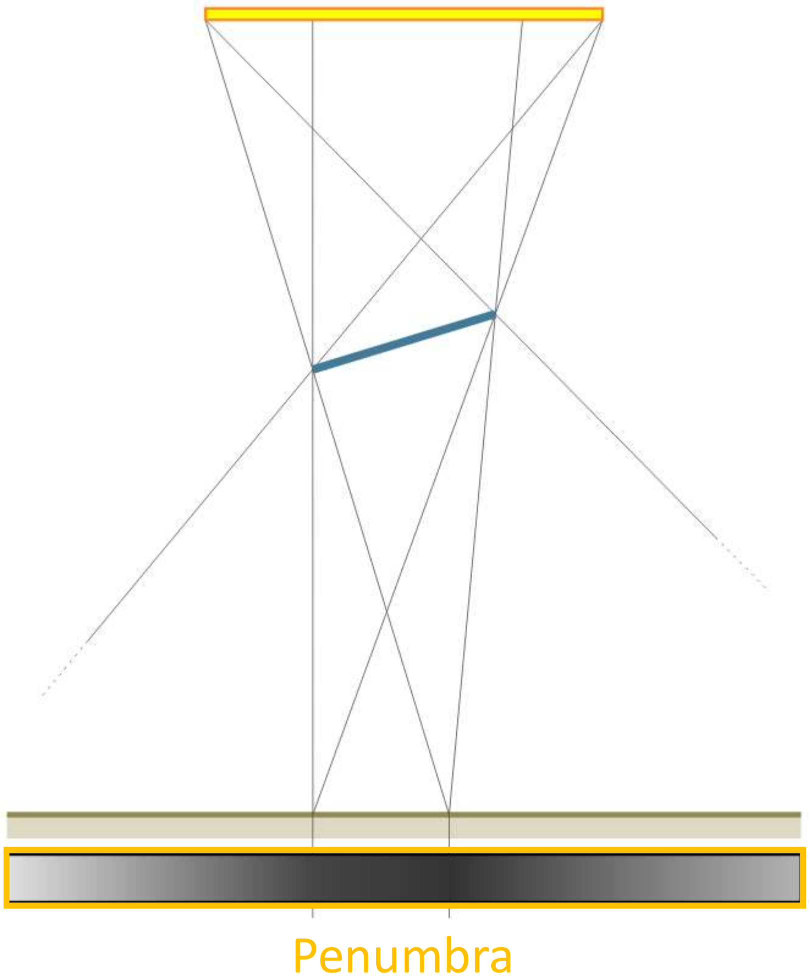


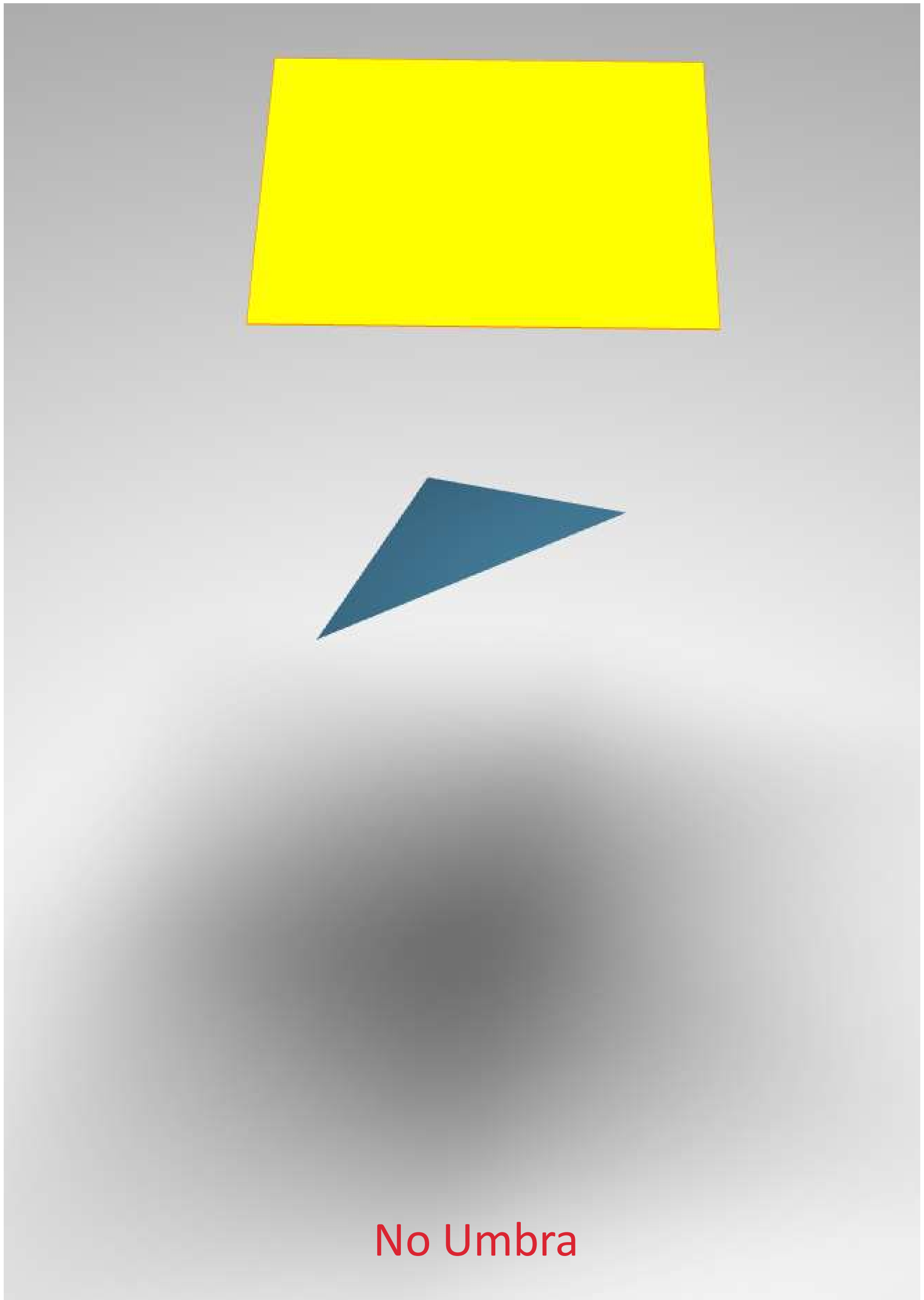
# Soft Shadows





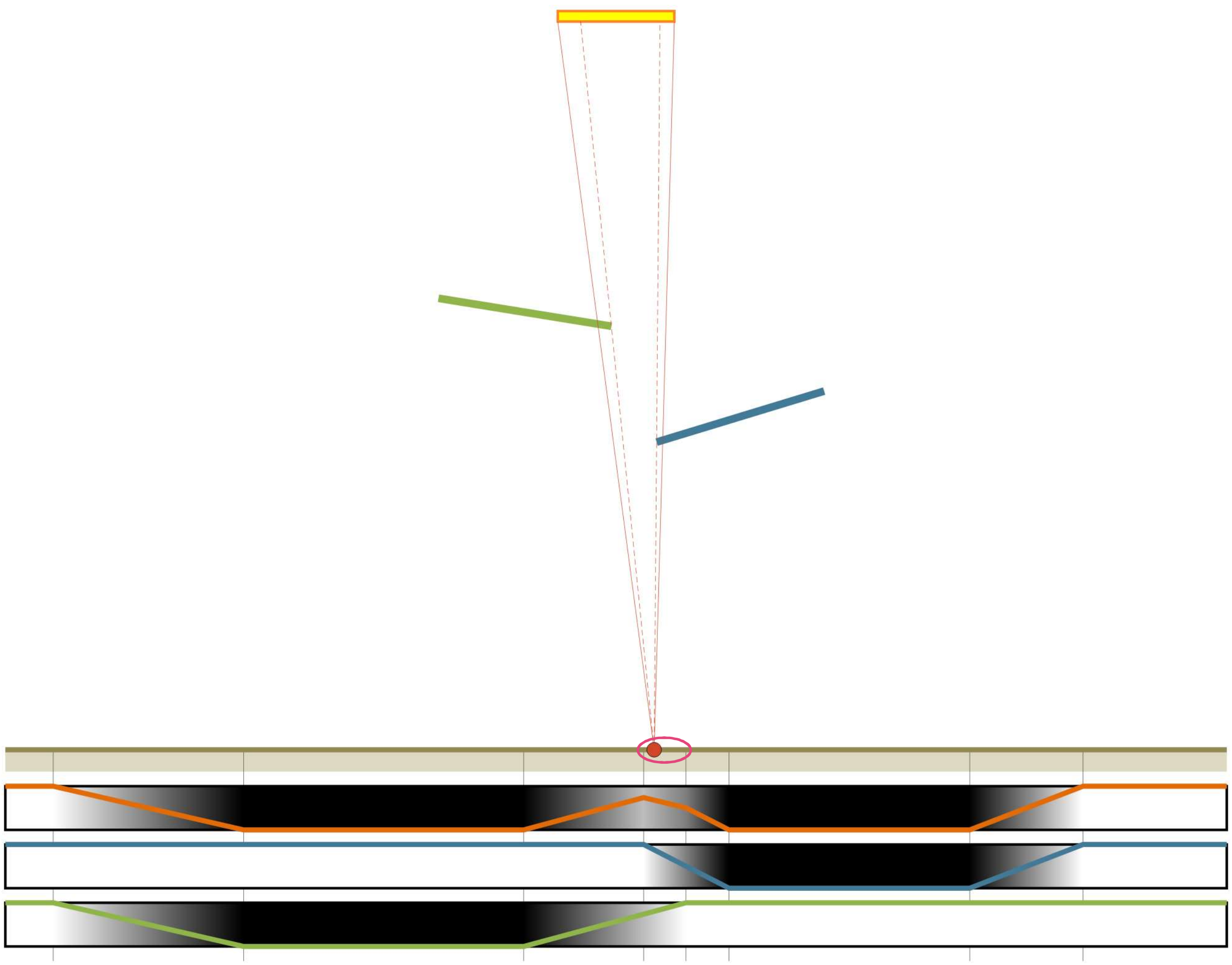
# Soft Shadows



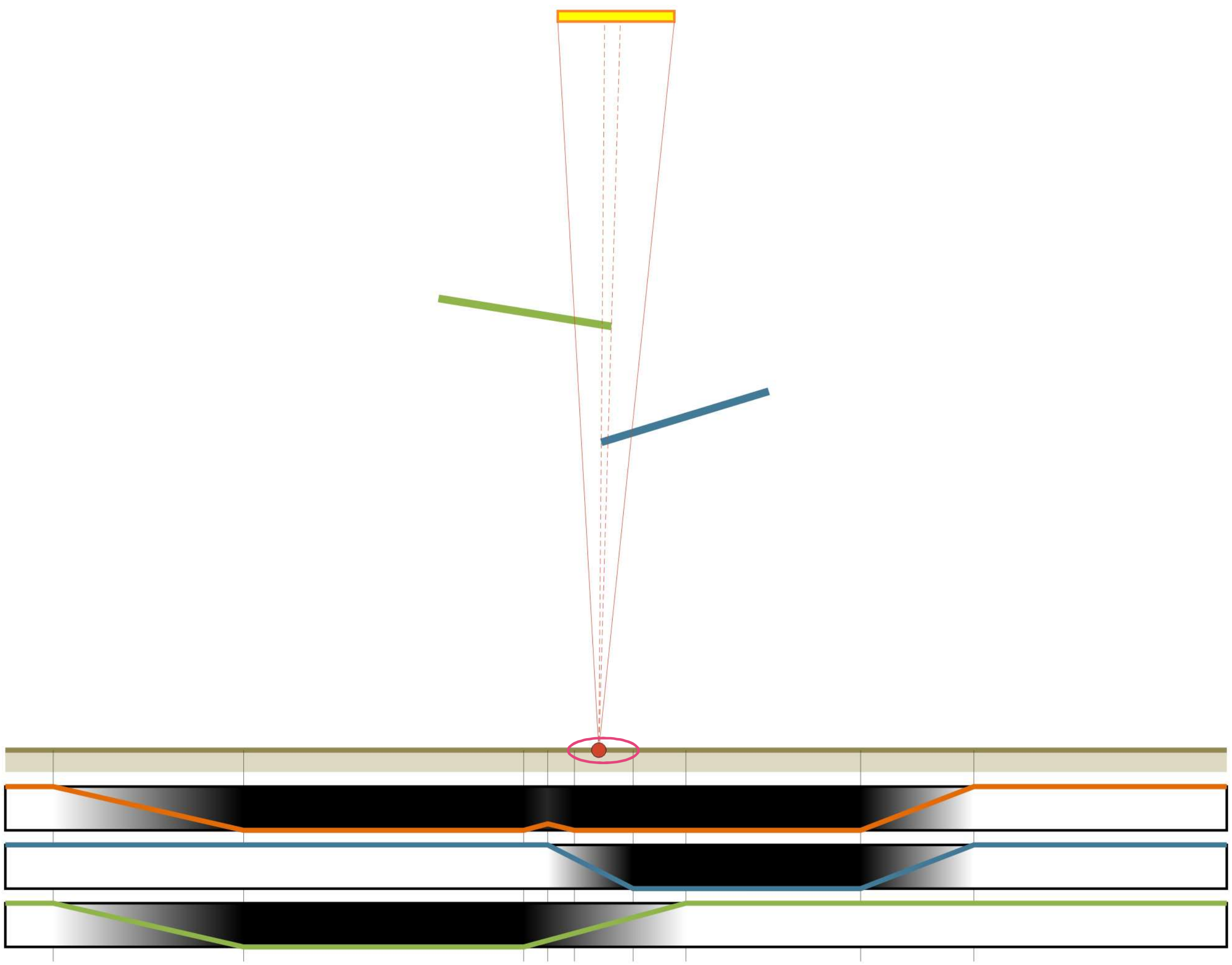



# Shadow Hardening on Contact

# Occluder Fusion

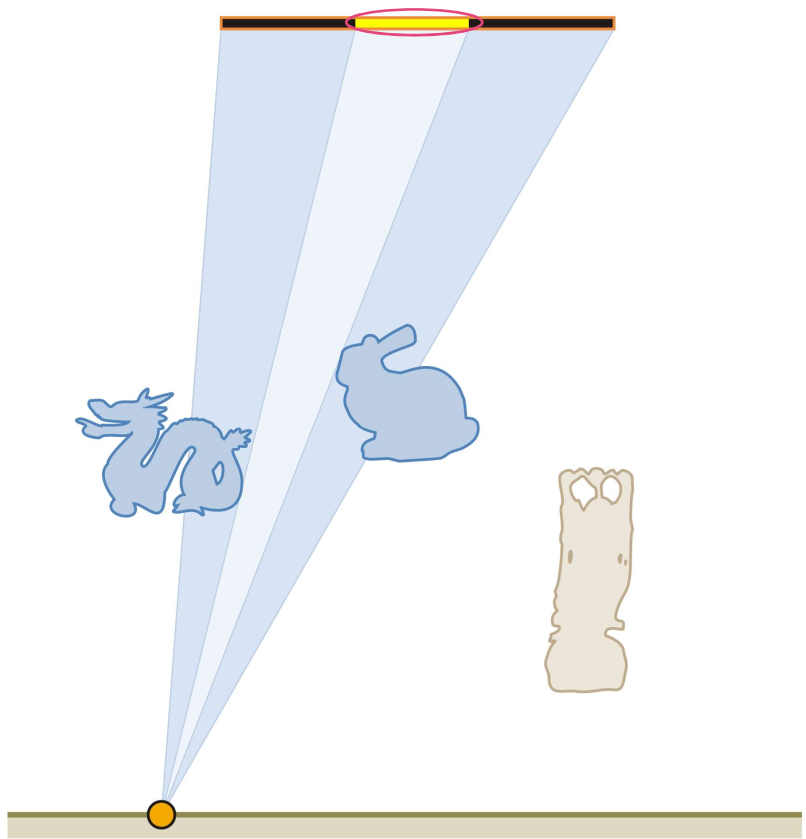


# Occluder Fusion



# Shadowing = Point–Region Visibility

- Task: Determine visible fraction of light source for each receiver sample



# Soft Shadows

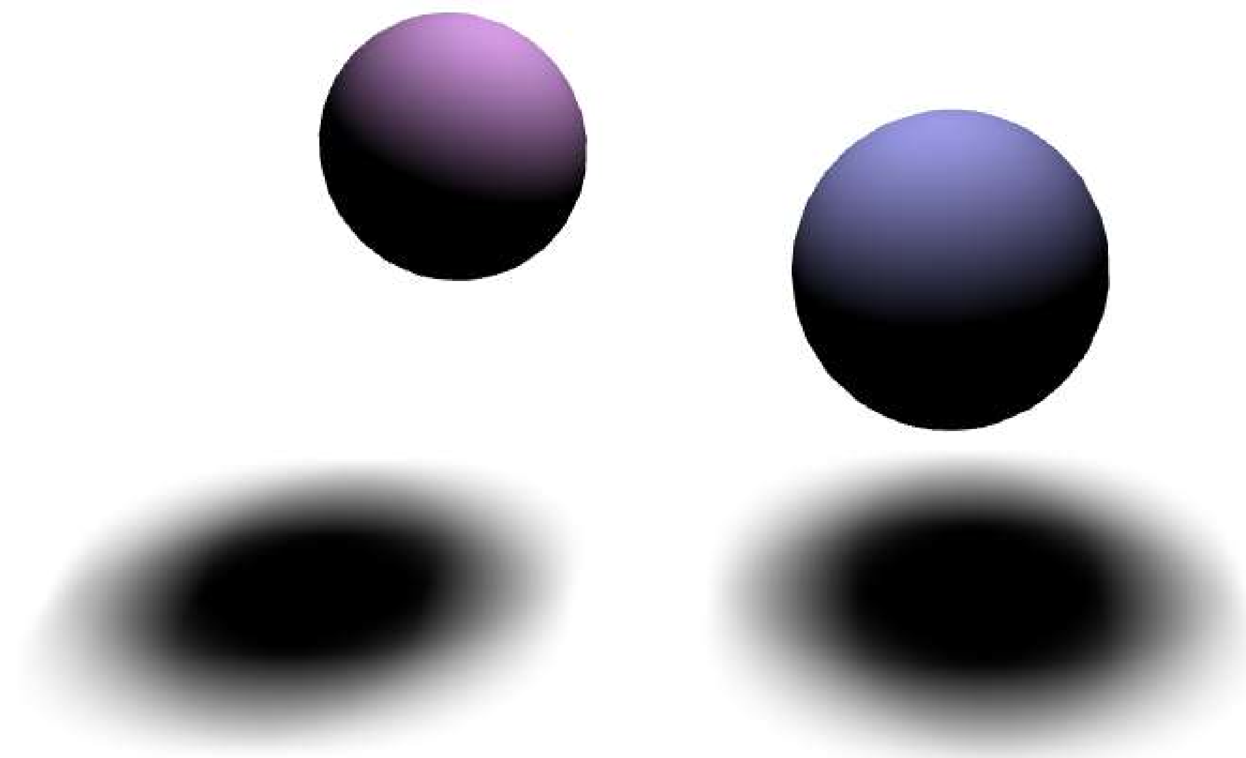
## Image-based Approaches

# Image-based Approaches

- Plausible faking by adapting hard shadows
- Adaptive blurring of hard shadow test results
  - Percentage-closer soft shadows
- Reconstructing and backprojecting occluders
  - Soft shadow mapping
- Utilize multiple shadow maps
- Occlusion textures

# Blurring of Hard Shadow Test Results

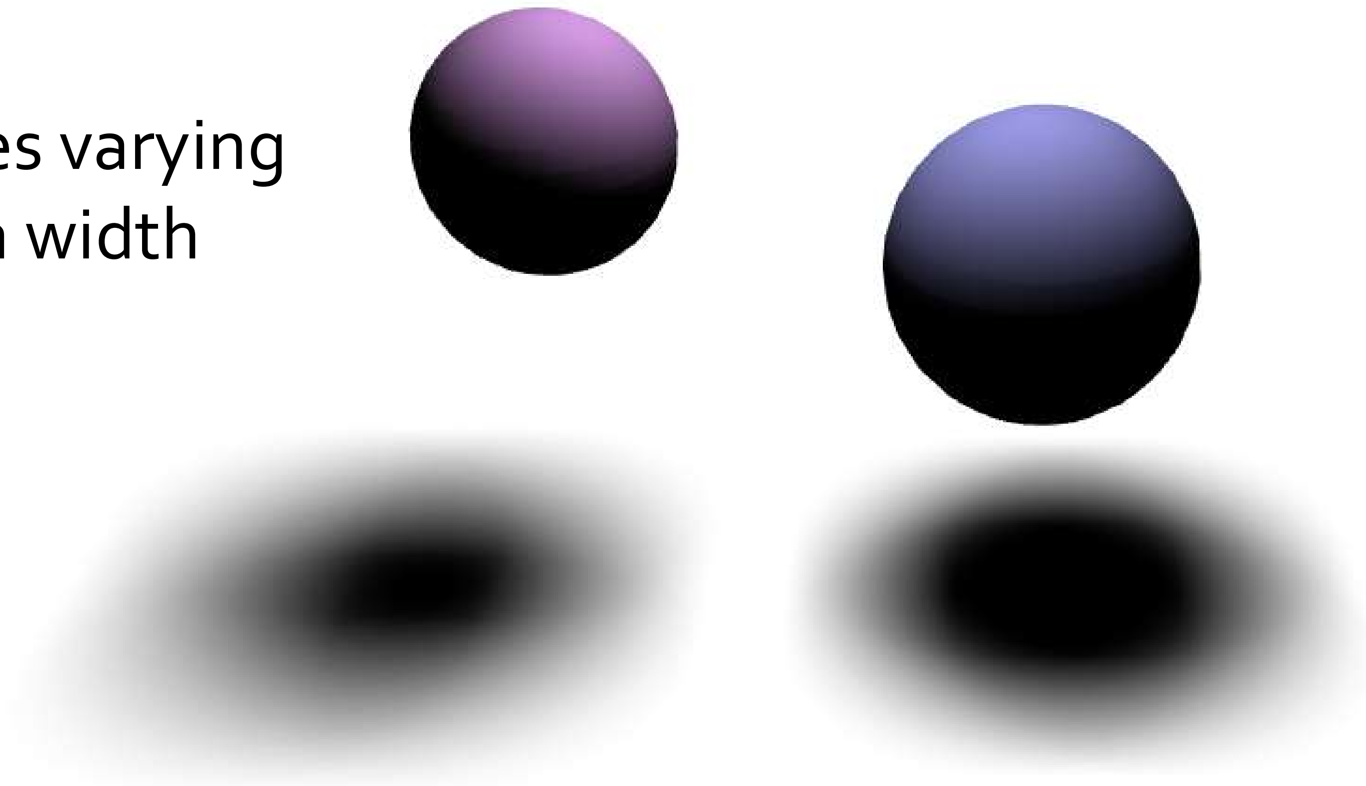
- Use any of the approaches presented before
- Yields a soft-shadow-like appearance



VSM, 512×512,
62×62

# Blurring of Hard Shadow Test Results

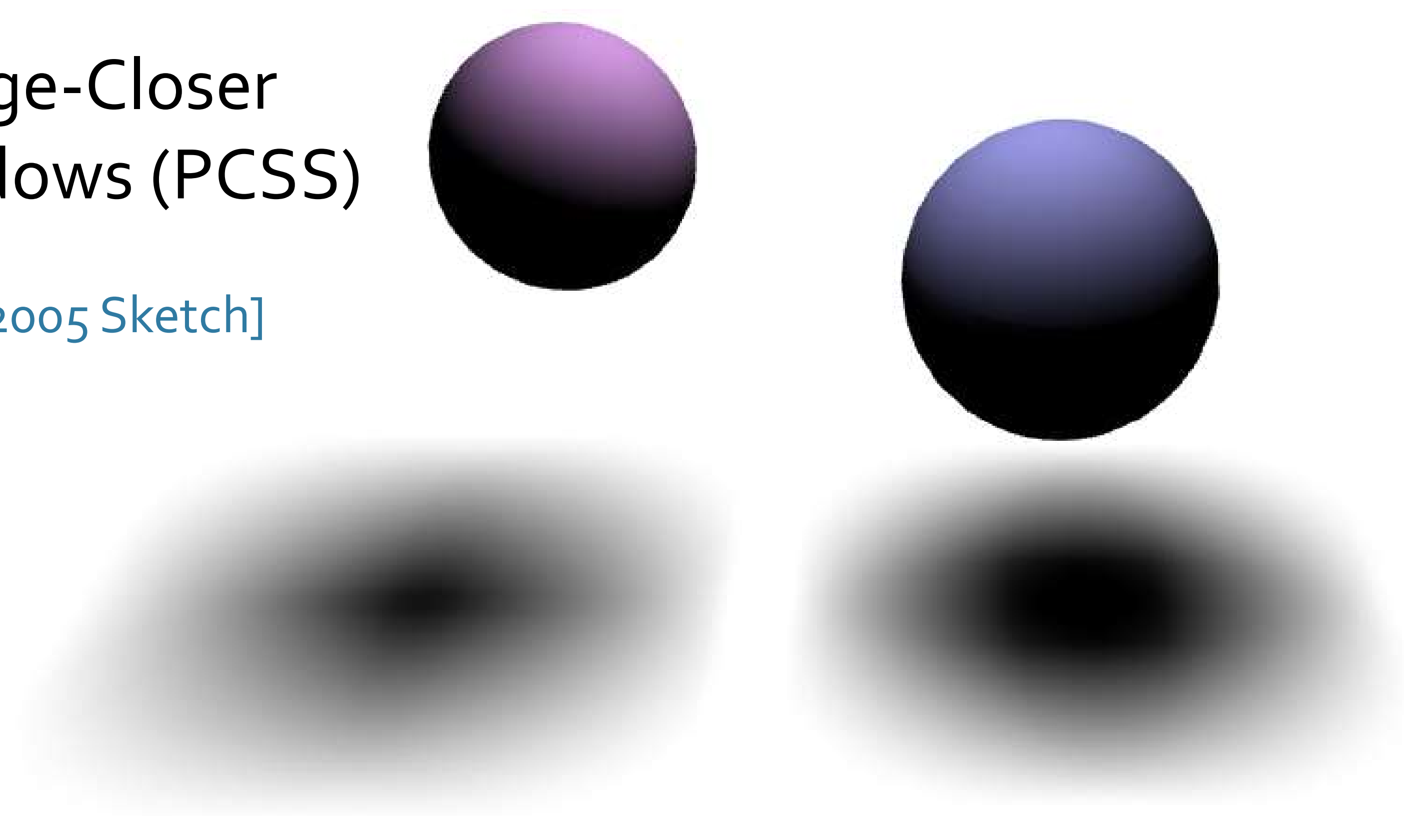
- Use any of the approaches presented before
- Yields a soft-shadow-like appearance
- But ignores varying penumbra width



Reference

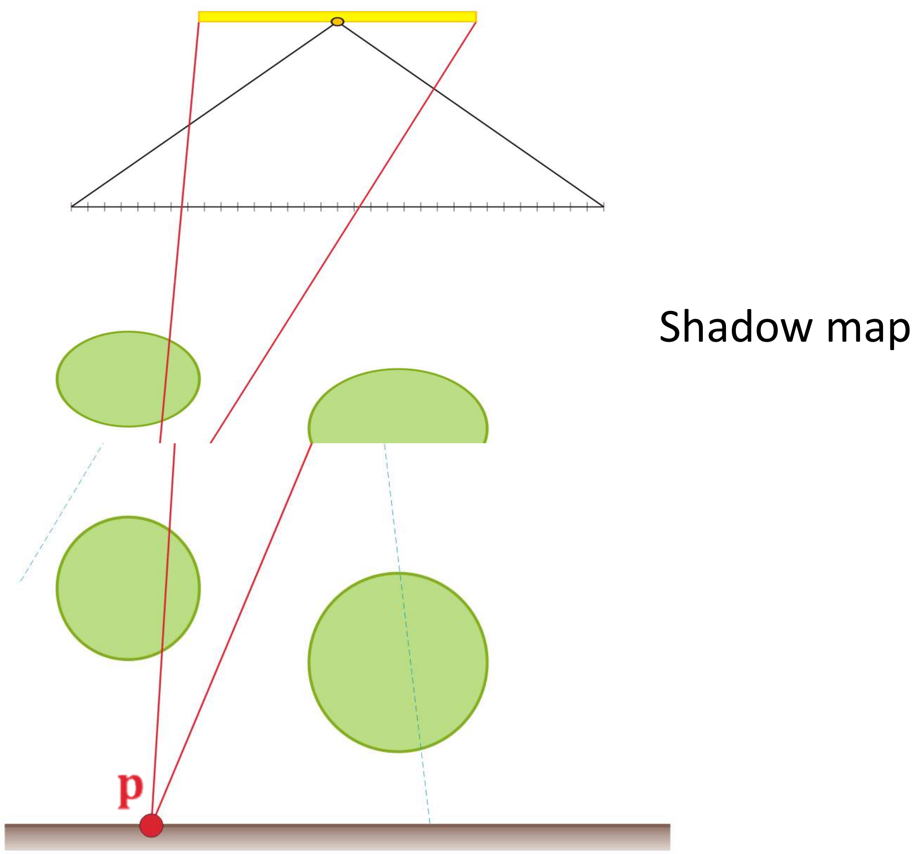
# Blurring of Hard Shadow Test Results

- Idea: Choose blur kernel size adaptively
  - But how?
- Percentage-Closer Soft Shadows (PCSS)
  [Fernando, SIGGRAPH 2005 Sketch]



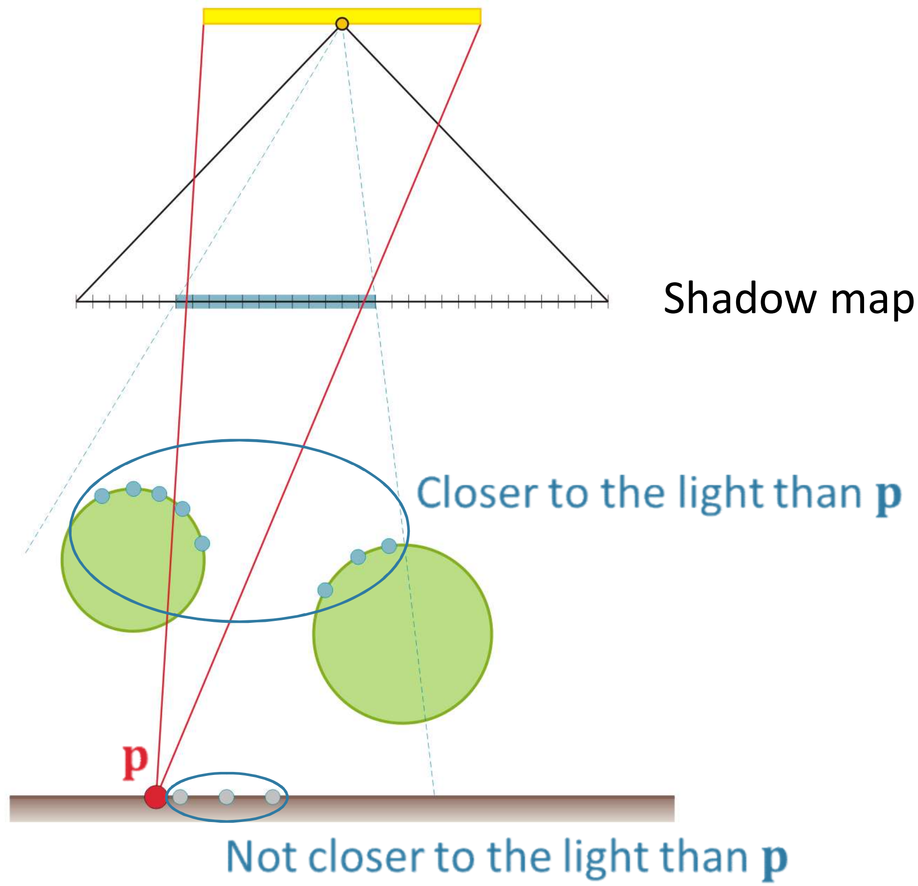
PCSS

# Percentage-Closer Soft Shadows



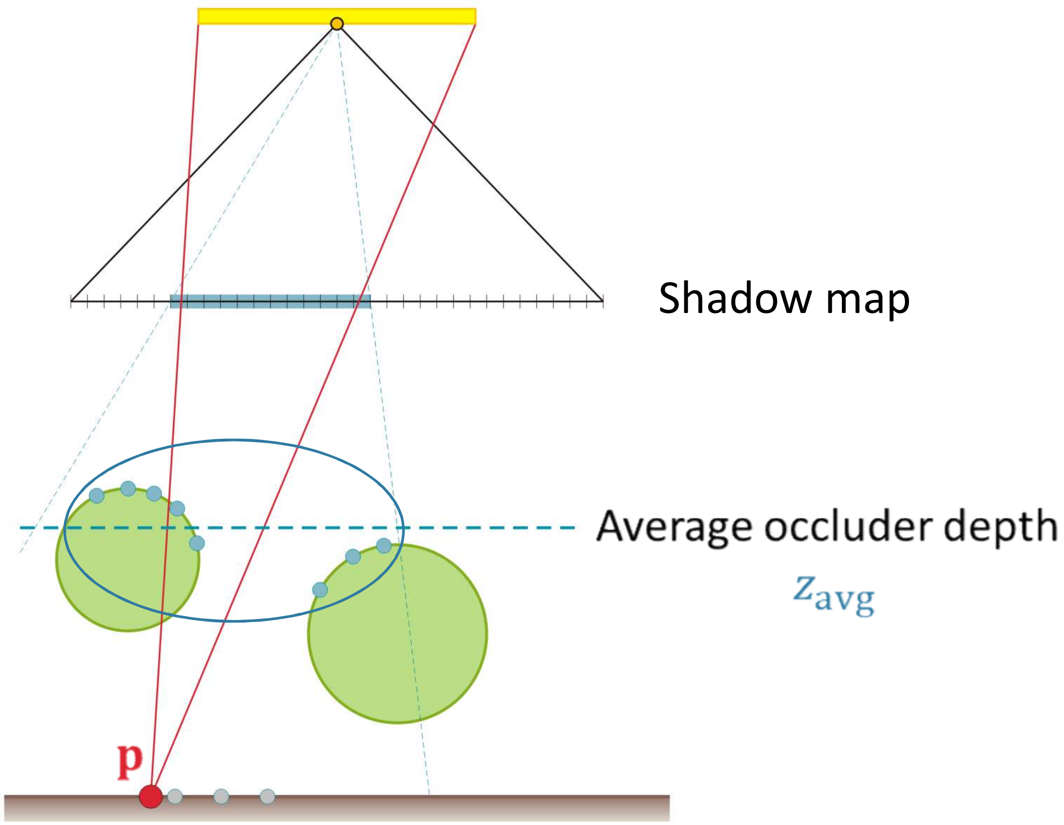

1. Blocker search

# Percentage-Closer Soft Shadows



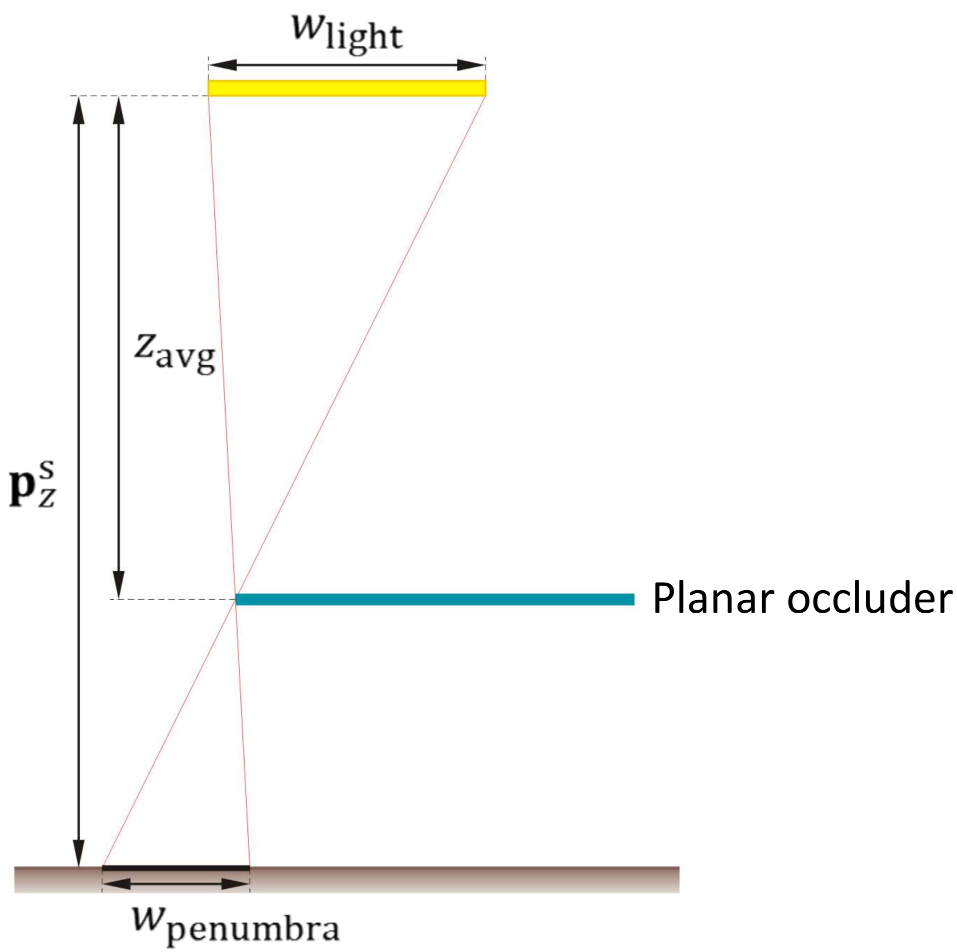

1. Blocker search

# Percentage-Closer Soft Shadows




1. Blocker search

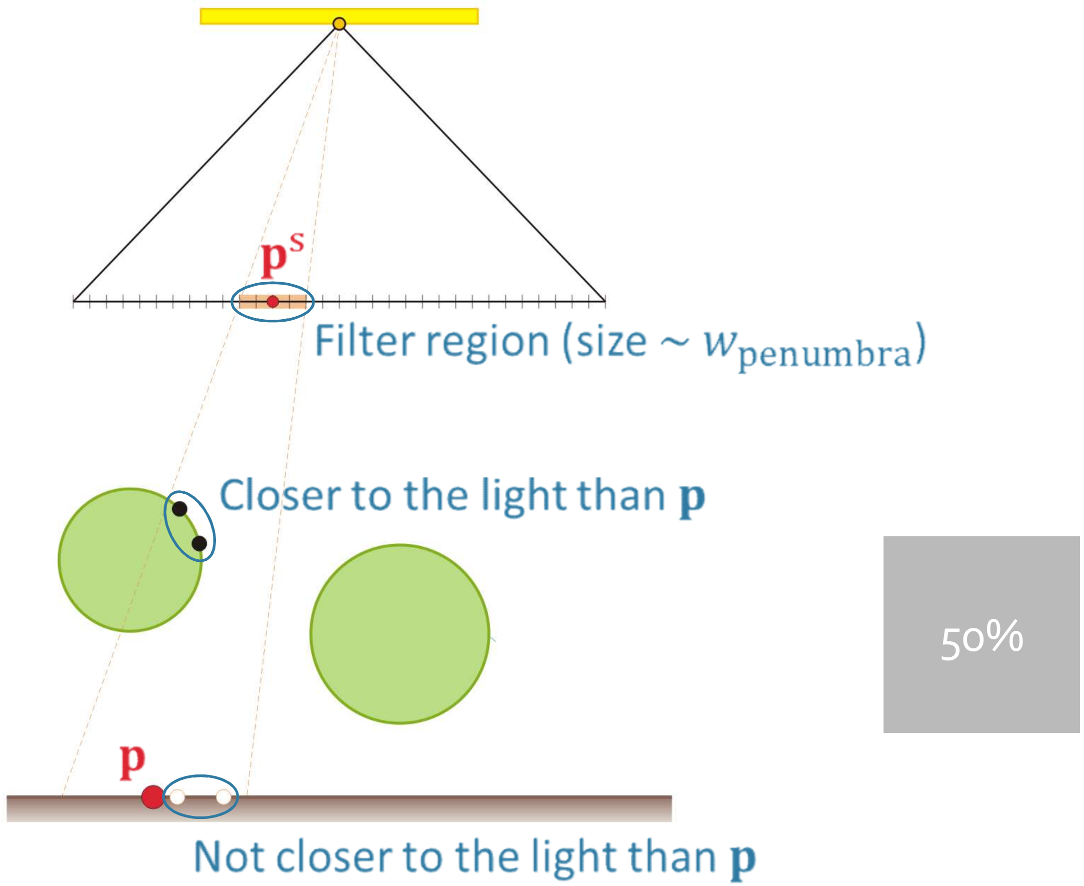
# Percentage-Closer Soft Shadows




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




3. Filtering: region proportional to penumbra width

# Percentage-Closer Soft Shadows

- Three steps
  - Blocker search
  - Penumbra width estimation
  - Filtering

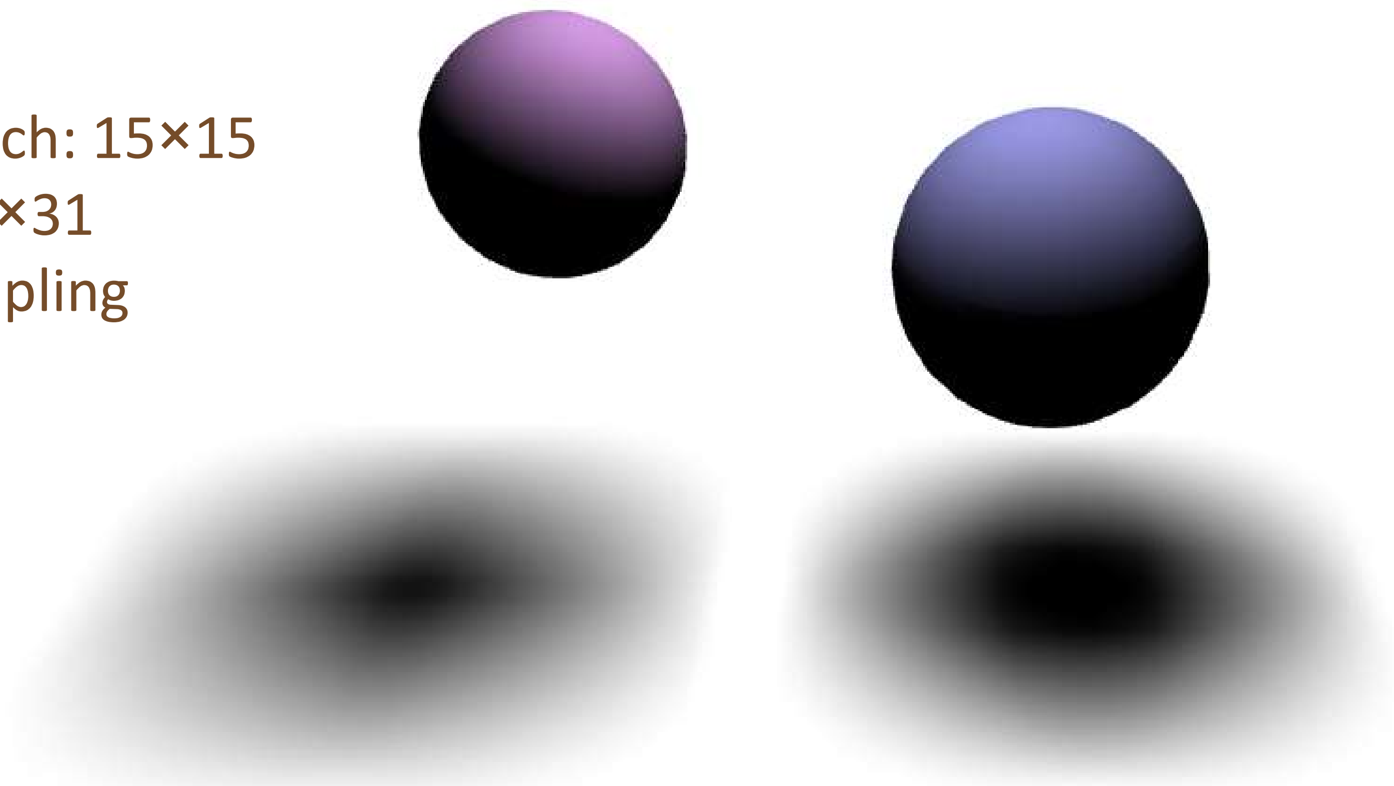
Two of them require many shadow map accesses!

- Speed-up approaches
  - Sub-sampling
  - Pre-filtering

# Percentage-Closer Soft Shadows

- Quality vs. number of shadow map samples

Blocker search: 15×15
Filtering: 31×31
Regular sampling

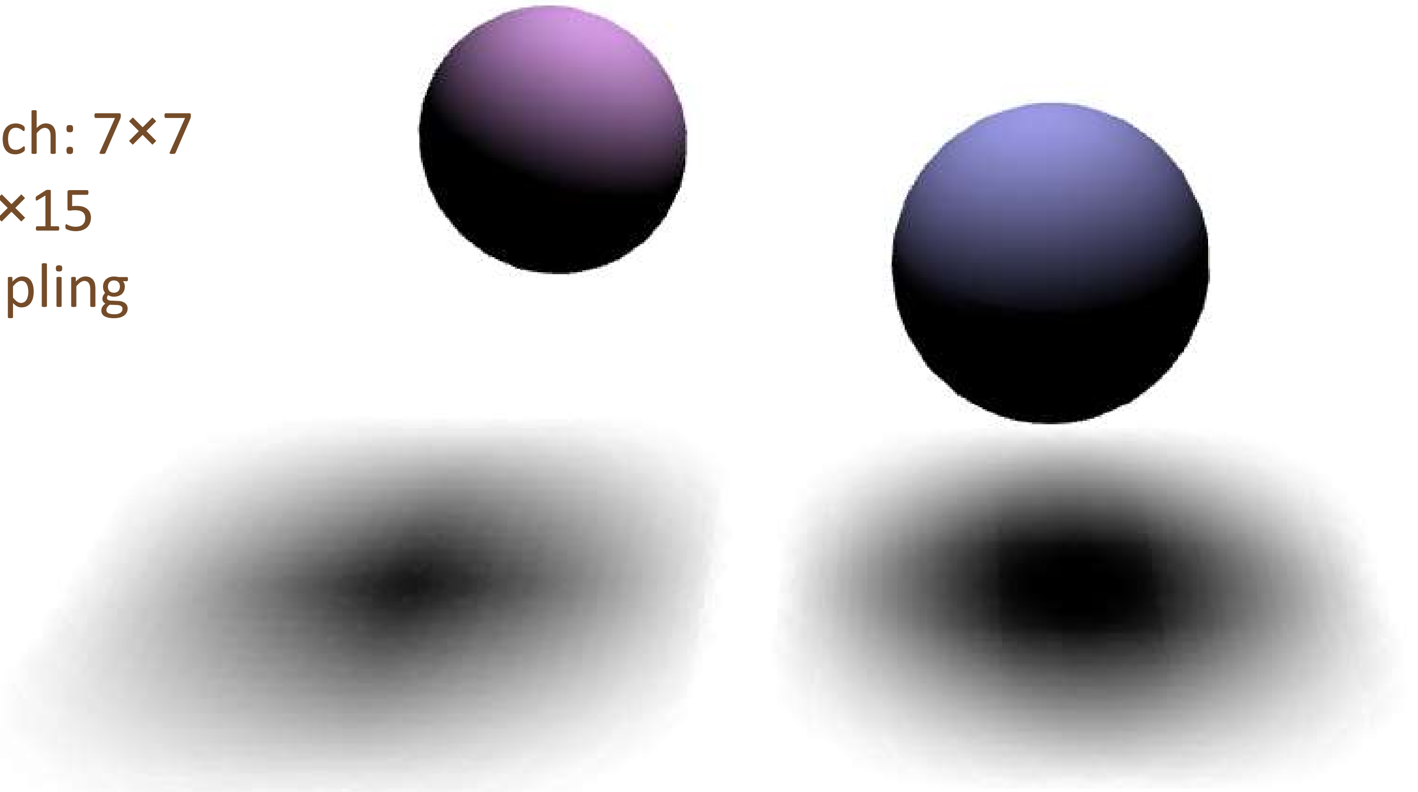


29 fps
(1024×1024, GeForce GTX 285)

# Percentage-Closer Soft Shadows

- Quality vs. number of shadow map samples

Blocker search: 7×7
Filtering: 15×15
Regular sampling

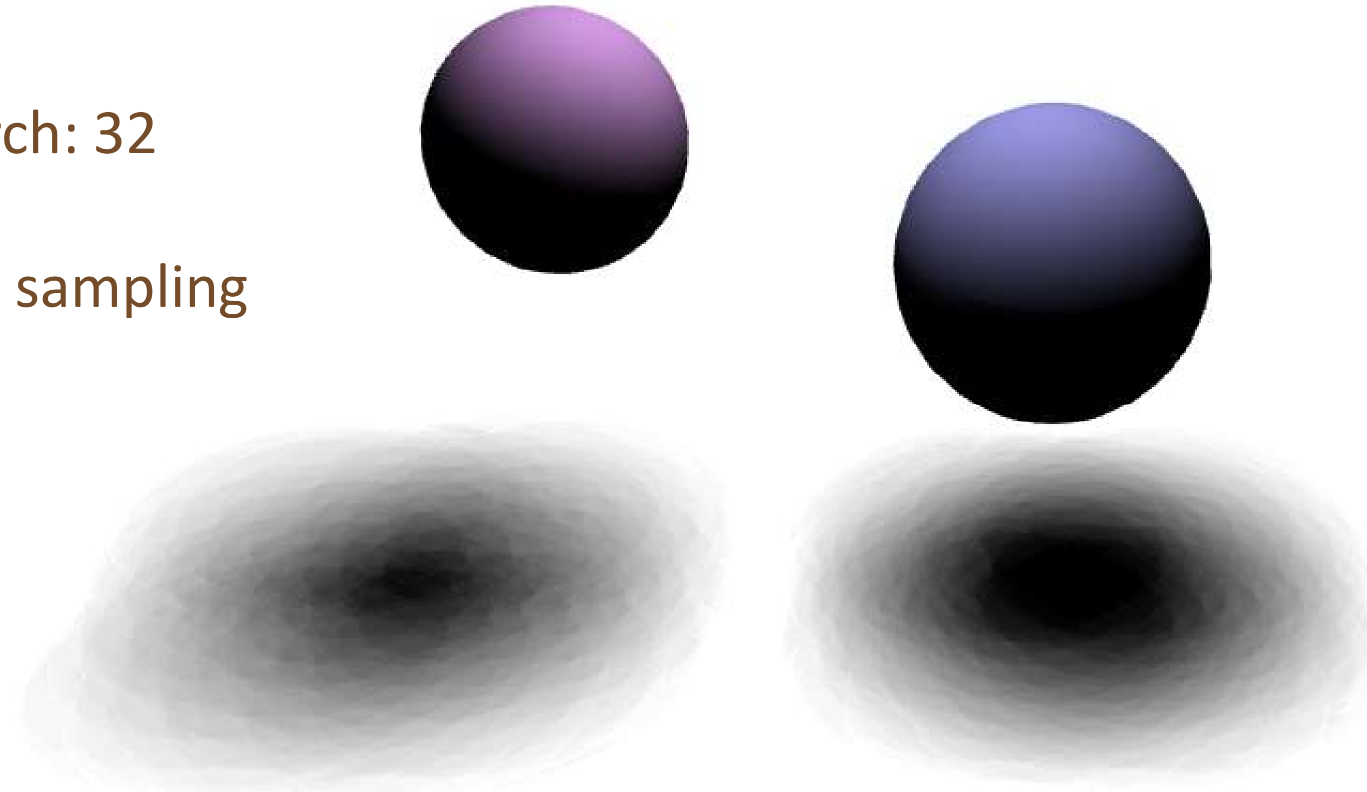


120 fps
(1024×1024, GeForce GTX 285)

# Percentage-Closer Soft Shadows

- Quality vs. number of shadow map samples

Blocker search: 32
Filtering: 64
Poisson disk sampling

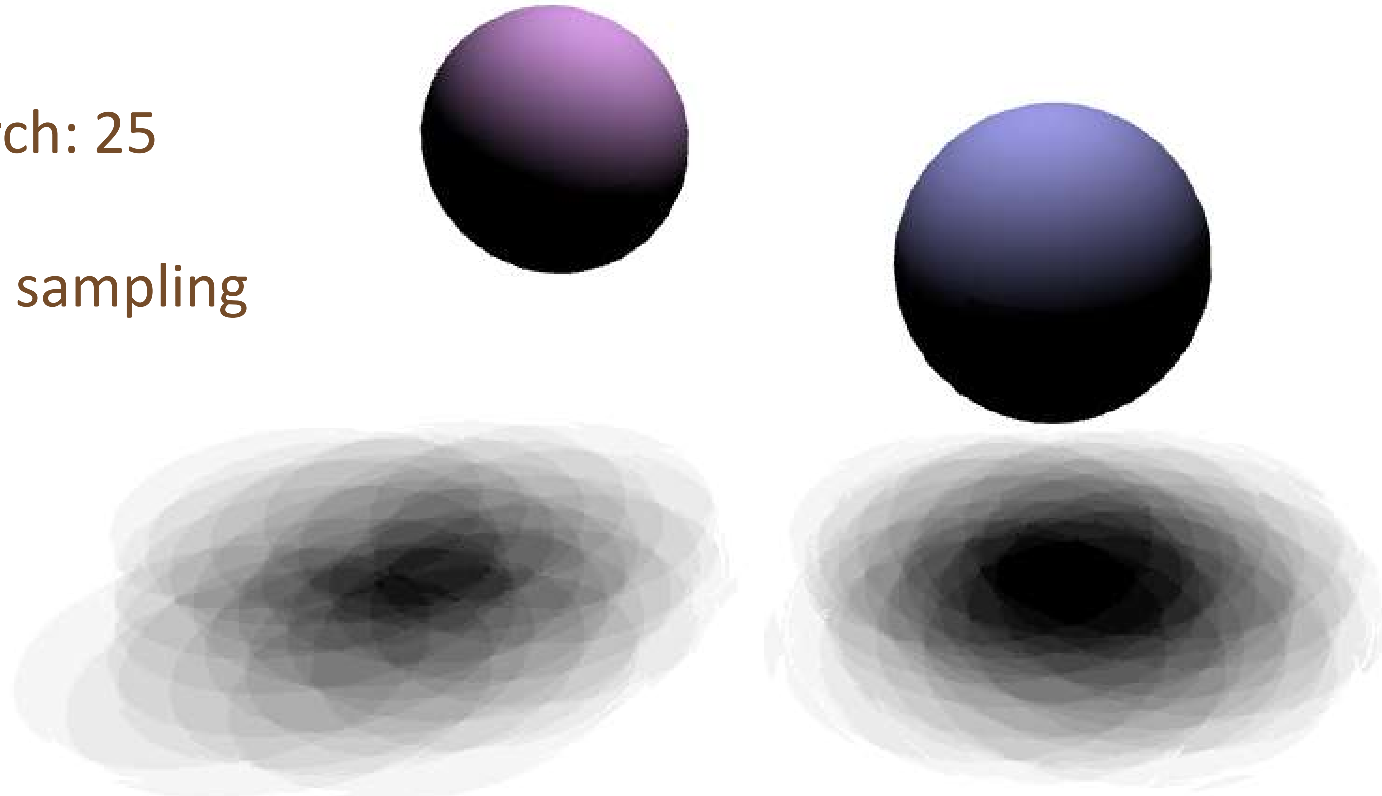


321 fps
(1024×1024, GeForce GTX 285)

# Percentage-Closer Soft Shadows

- Quality vs. number of shadow map samples

Blocker search: 25
Filtering: 25
Poisson disk sampling



519 fps
(1024×1024, GeForce GTX 285)

# Pre-Filtering

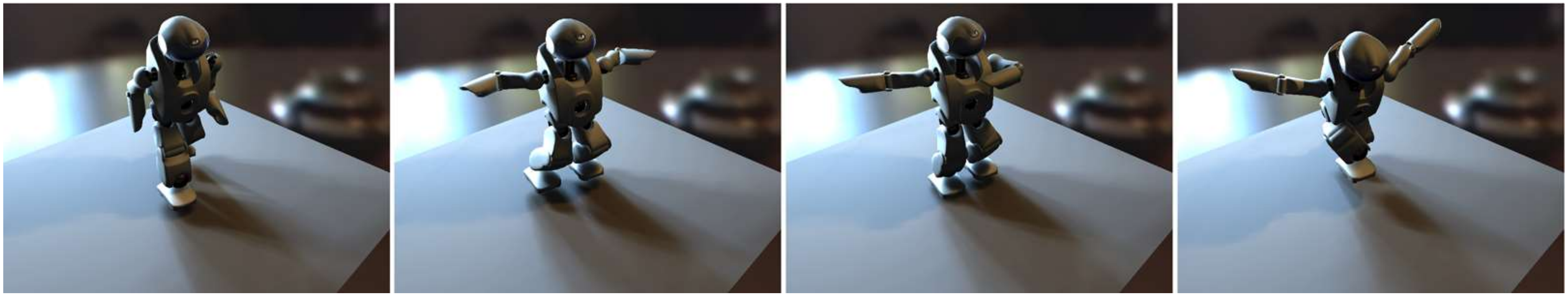
- Filtering step does just percentage-closer filtering
  - Using alternative shadow map representations (like VSM or CSM) allows for pre-filtering
  - → blurring reduces to a single texture fetch

But how to support adaptive filter region sizes?

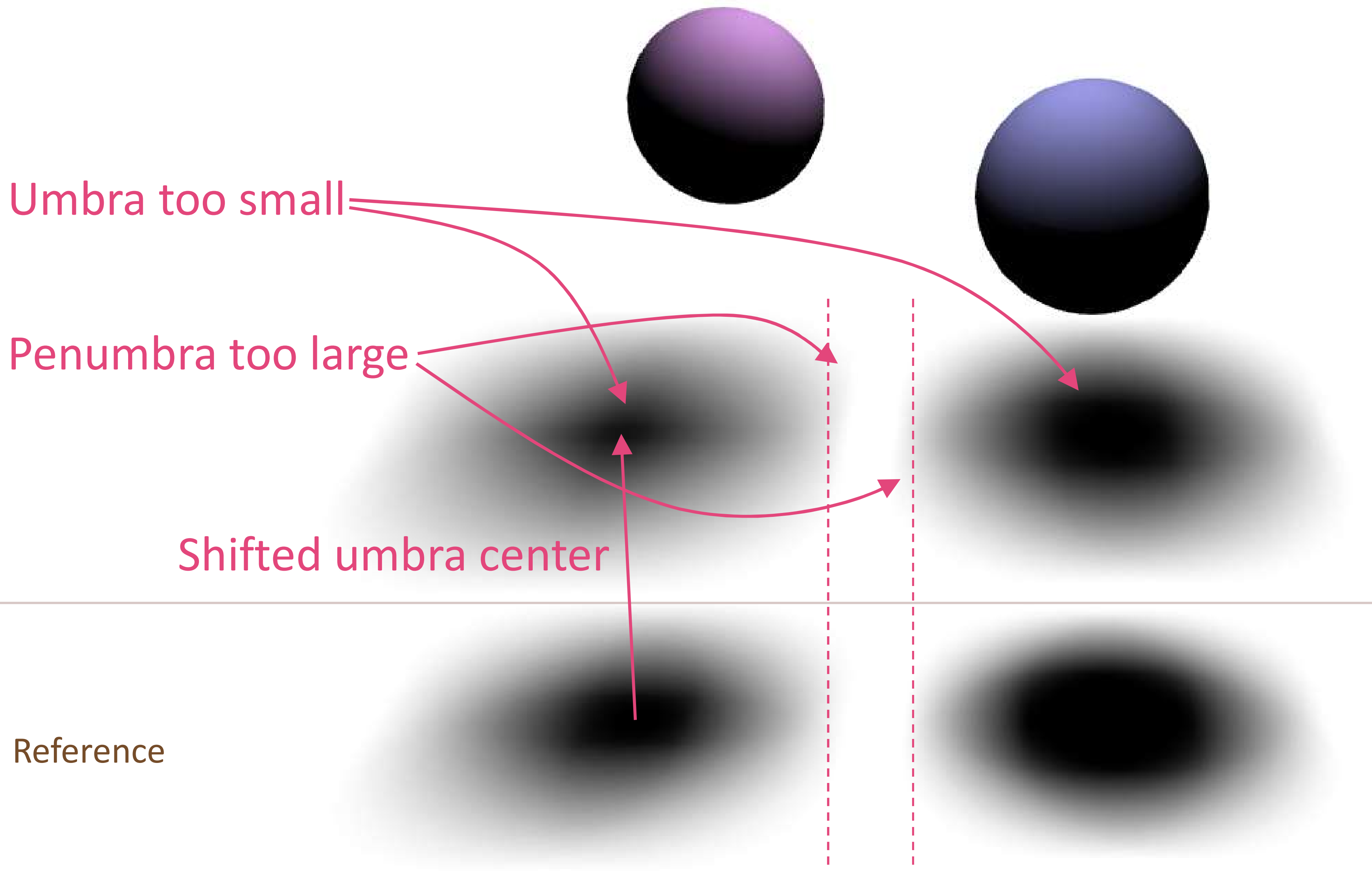
- Mipmapping (or alternatively N-buffers)
  - Store results for several discrete filter sizes and interpolate
- Summed-area table
  - Supports arbitrary rectangular box filter kernels

# Pre-Filtering

- Blocker search can also be sped up with pre-filtering
- Convolution Soft Shadows
  [Annen et al., SIGGRAPH 2008]
  - Observation: Averaging the depth of shadow map samples closer to the light can be expressed as a convolution
  - Hence approach analogous to CSM is possible

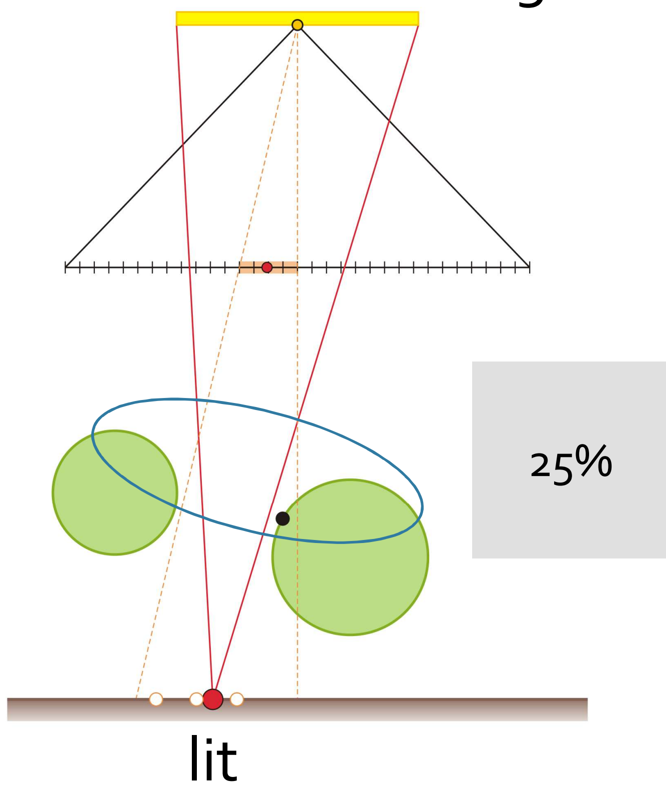


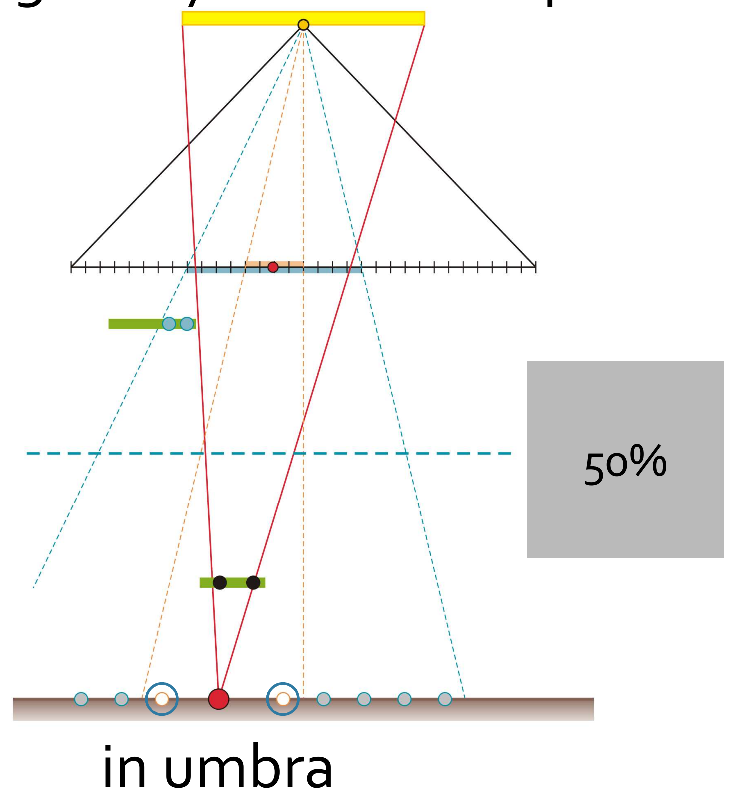
# Percentage Closer Soft Shadows - Problems



# Percentage Closer Soft Shadows - Problems

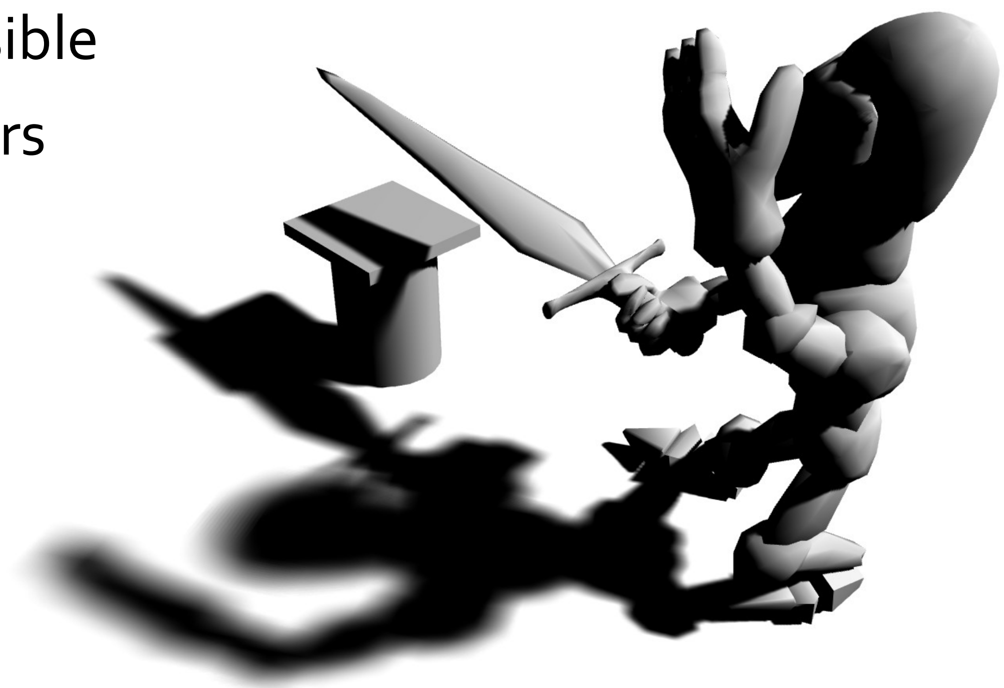
- Main sources of physical incorrectness
  - Single planar occluder assumption
  - Classification as light blocking solely based on depth test



# 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



# Occluder Backprojection

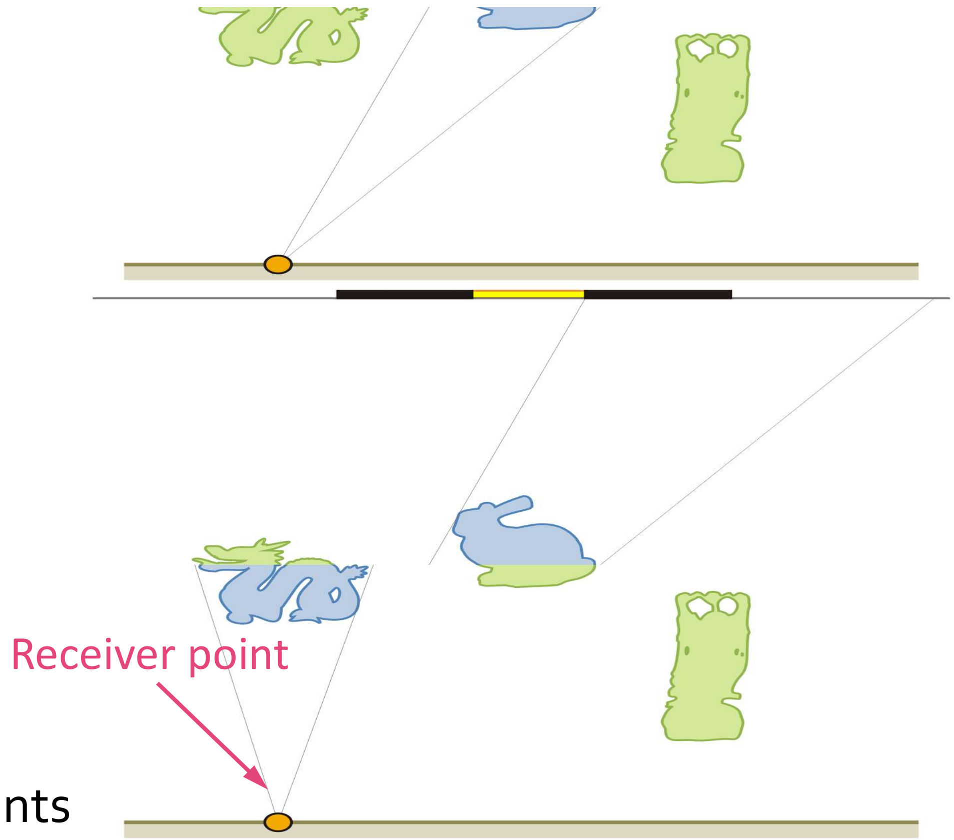
- For each (relevant) occluder
  - Project it onto light source
  - Determine covered light area
  - Aggregate this occlusion information
- Gathering

  For all receiver points

  For all potential occluders
- Scattering

  For all occluders

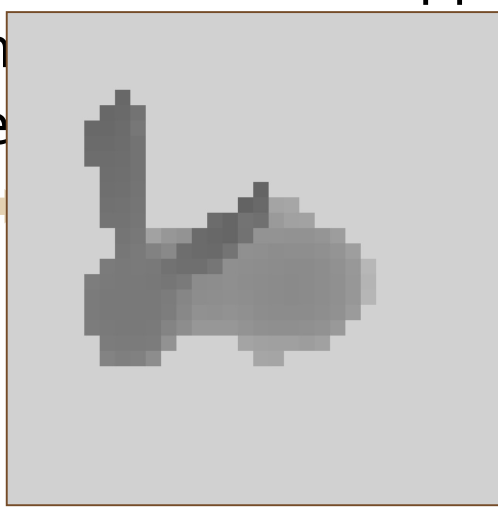
  For all affected receiver points

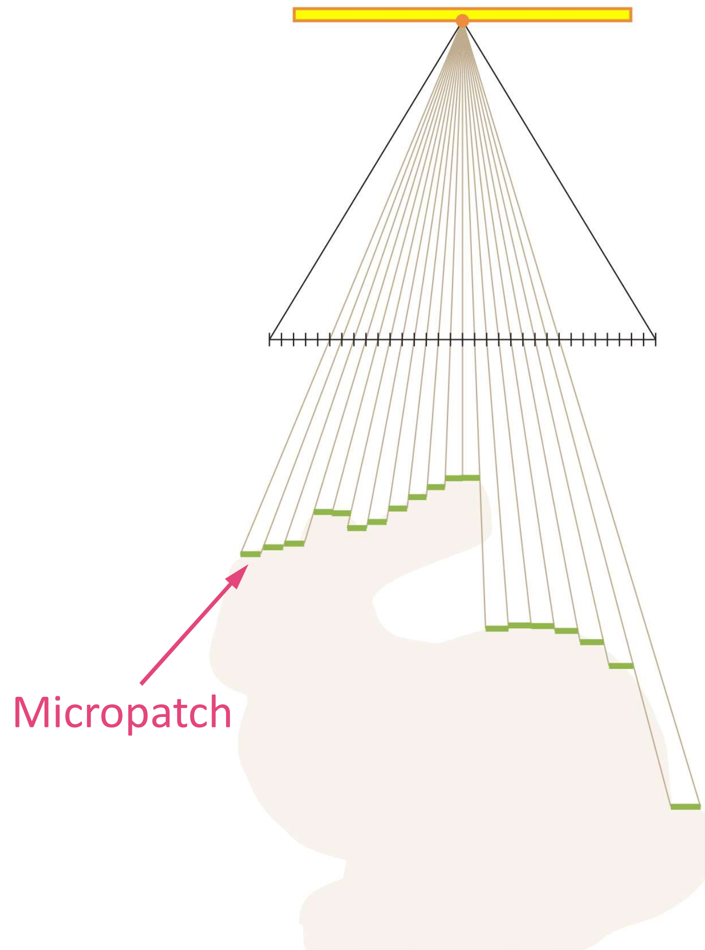


# Soft Shadow Mapping

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

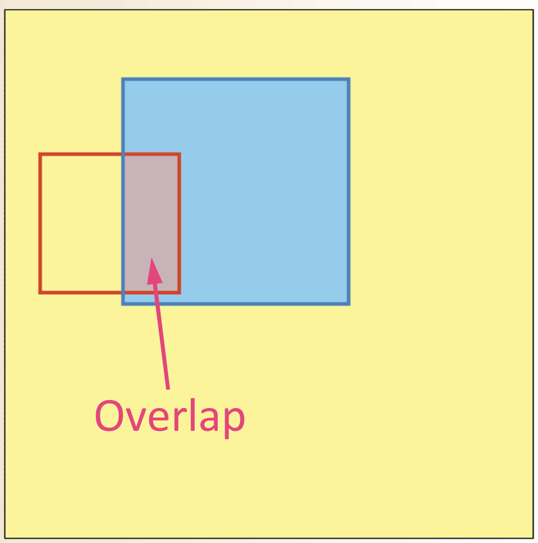
- Approximate (subset of) occluder geometry
- Generate shadow map (from light's center)
- Derive occluder approx- in ting le ce

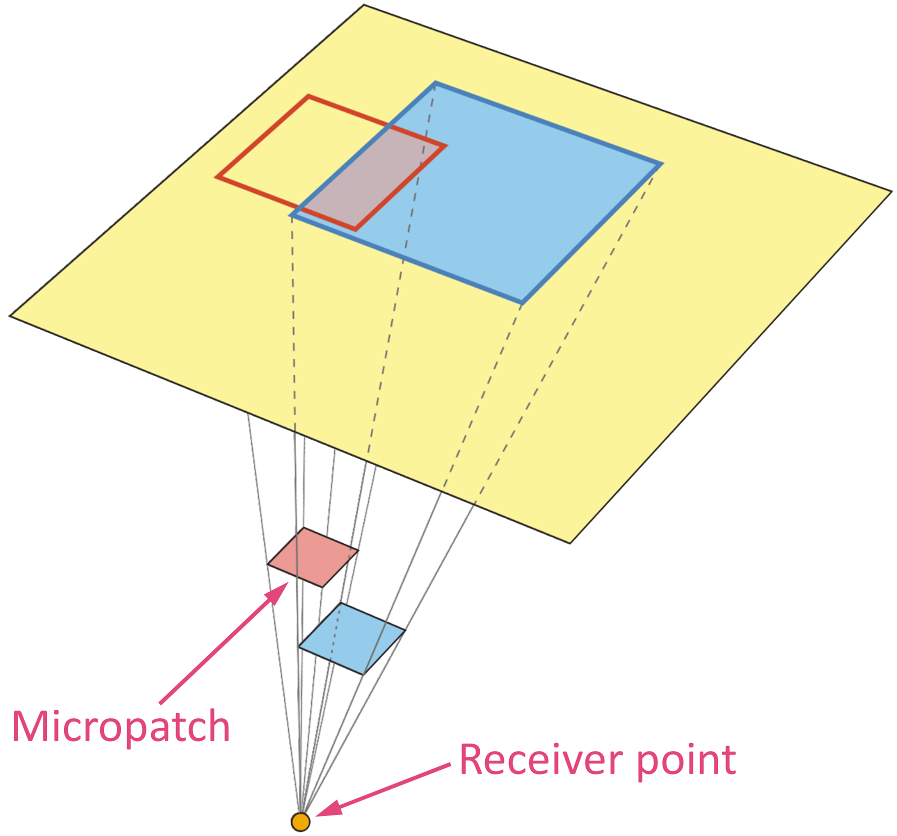




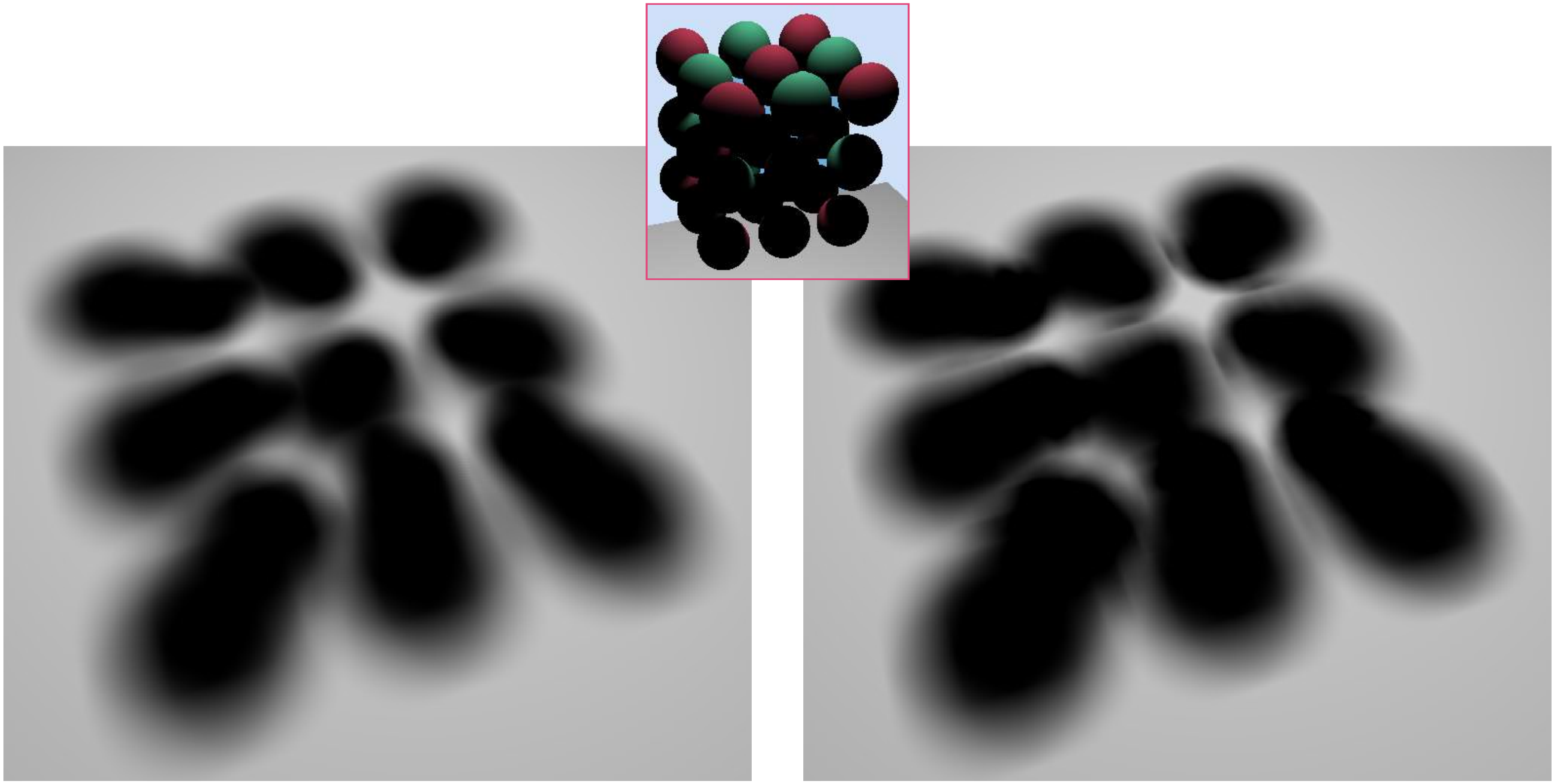
# Soft Shadow Mapping

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



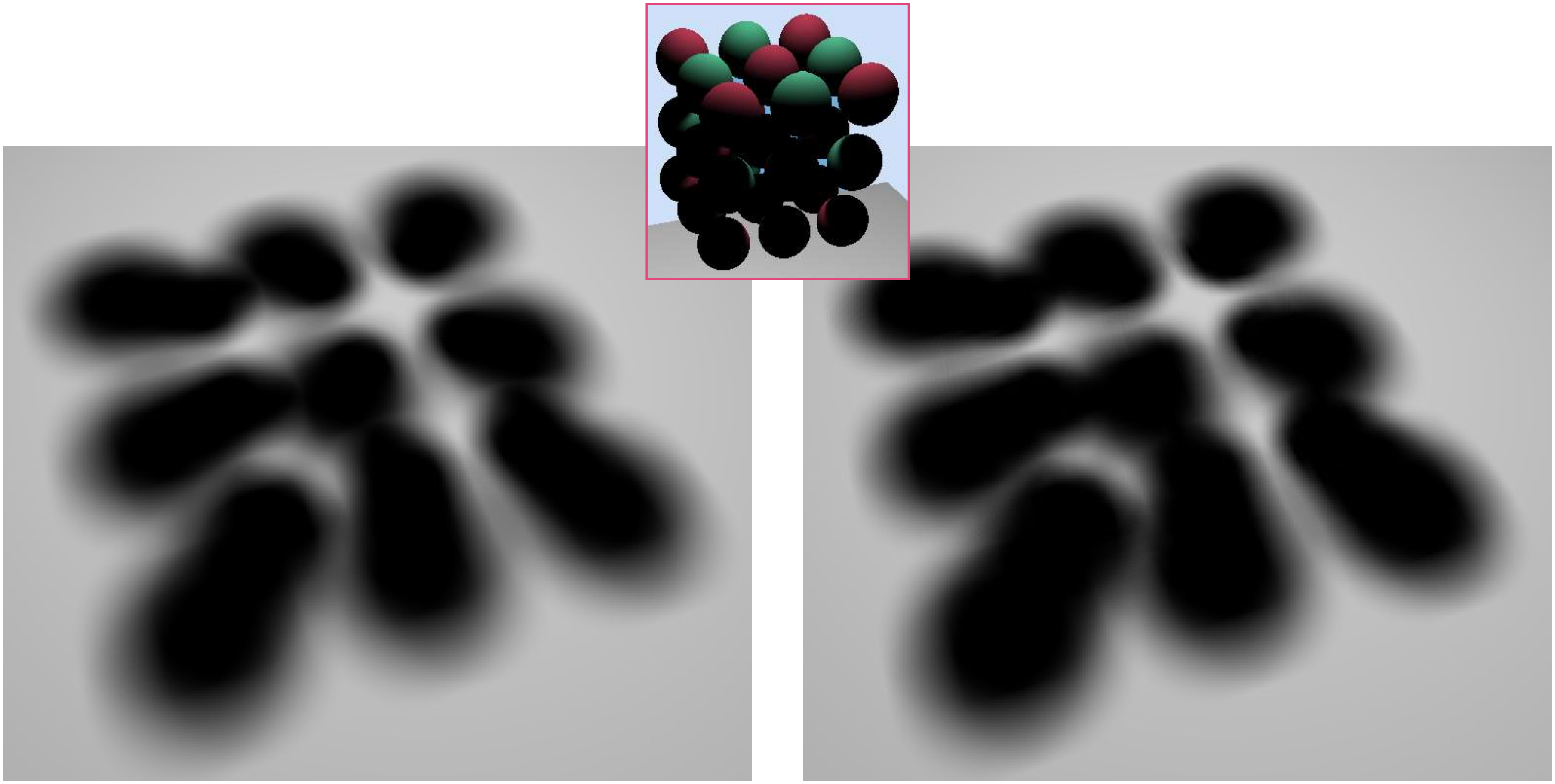
# Overlapping Artifacts



Reference

Area accumulation
(without accounting for overlaps)

# Overlapping Artifacts



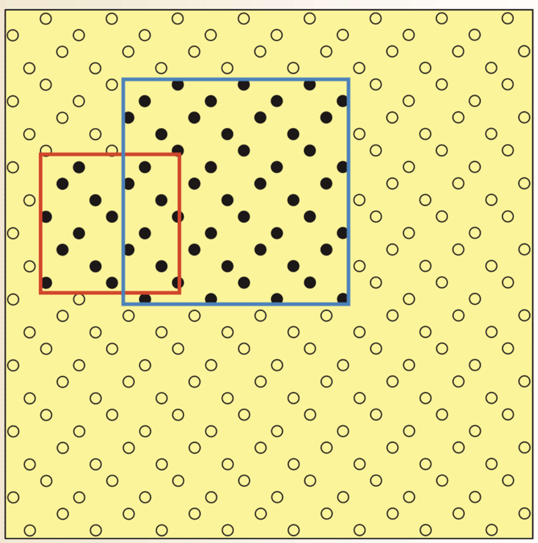
Reference

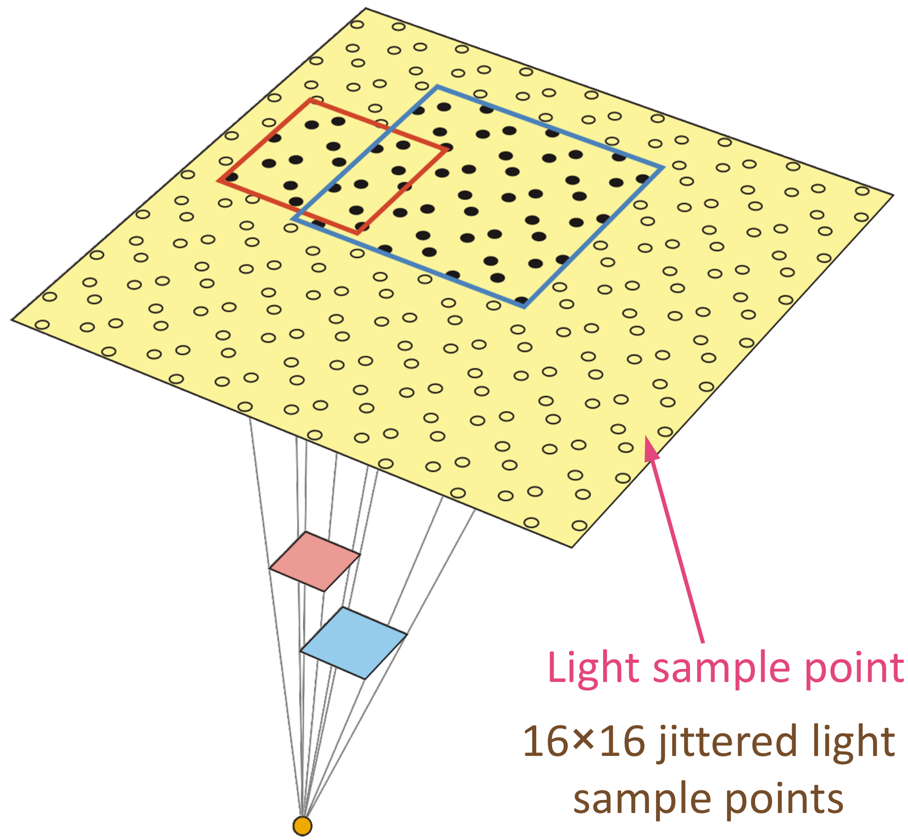
Occlusion bitmasks
(with correct overlap handling)

# Occlusion Bitmasks

[Schwarz & Stamminger, EG 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

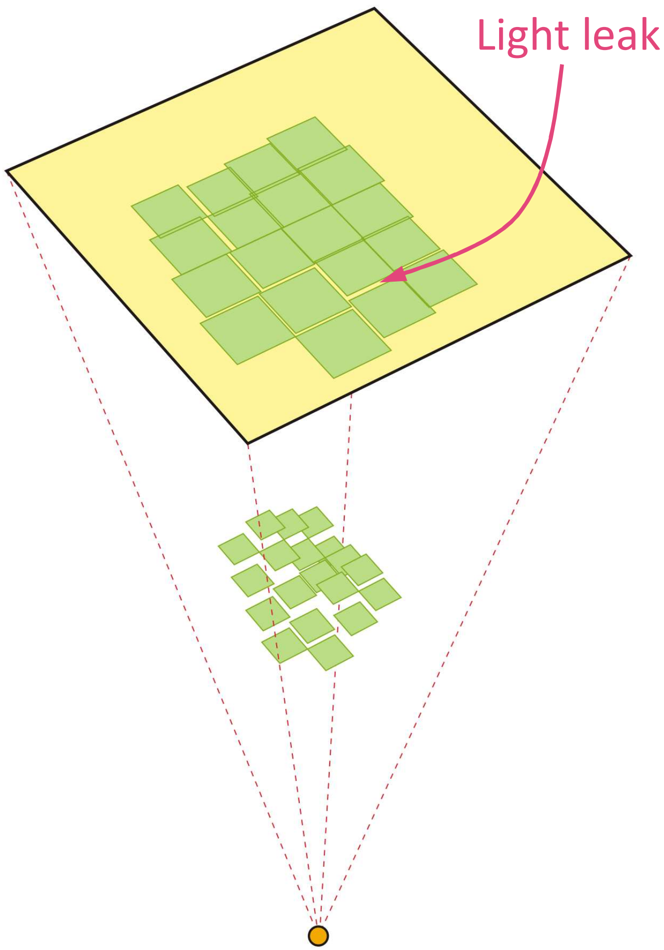


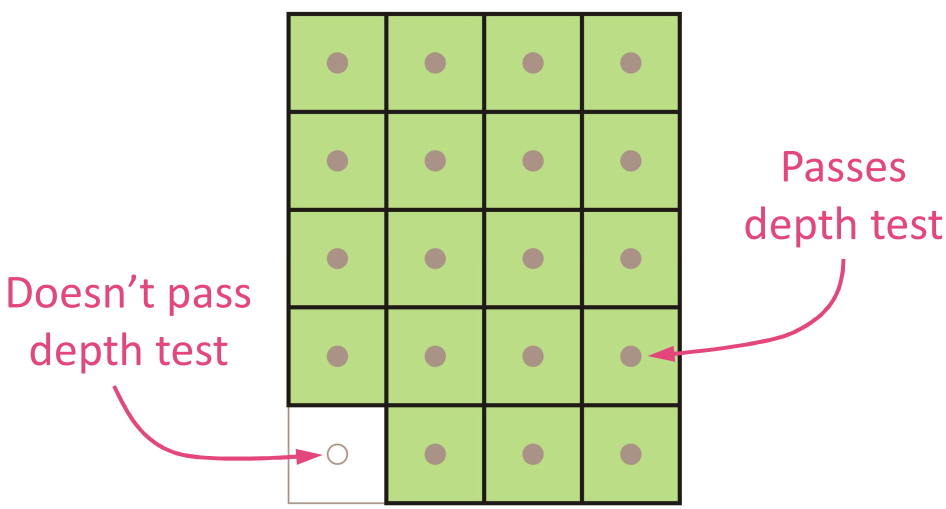


# Micropatch Light Leaks

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

- Lexel-sized rectangles constructed around each unprojected sample that passes depth test

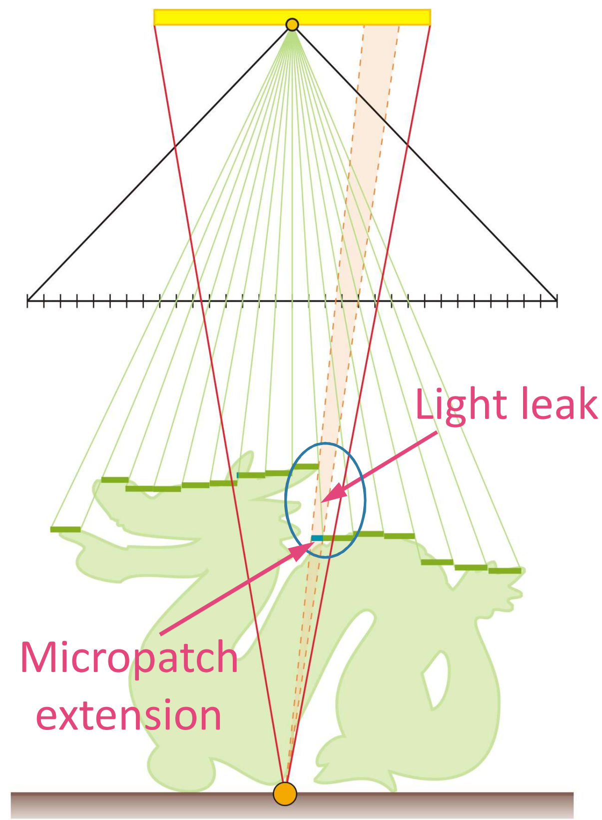



Texture space

# Micropatches

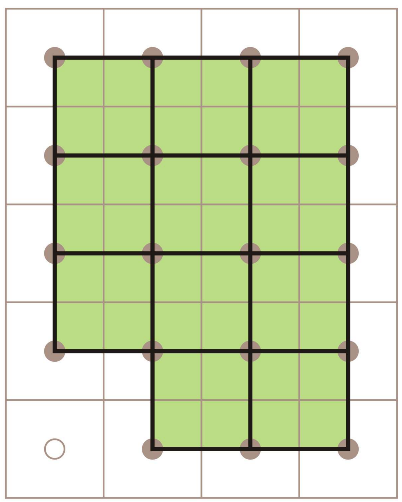
- Suffer from light leaks
  - Requires gap filling,
    e.g. by extending micropatches
- Piecewise-constant approximation
  - Prone to surface-acne artifacts
- Frequent occluder overestimation
  - But helps capturing fine structures
- Backprojection is
  axis-aligned rectangle
  - Simple and fast processing

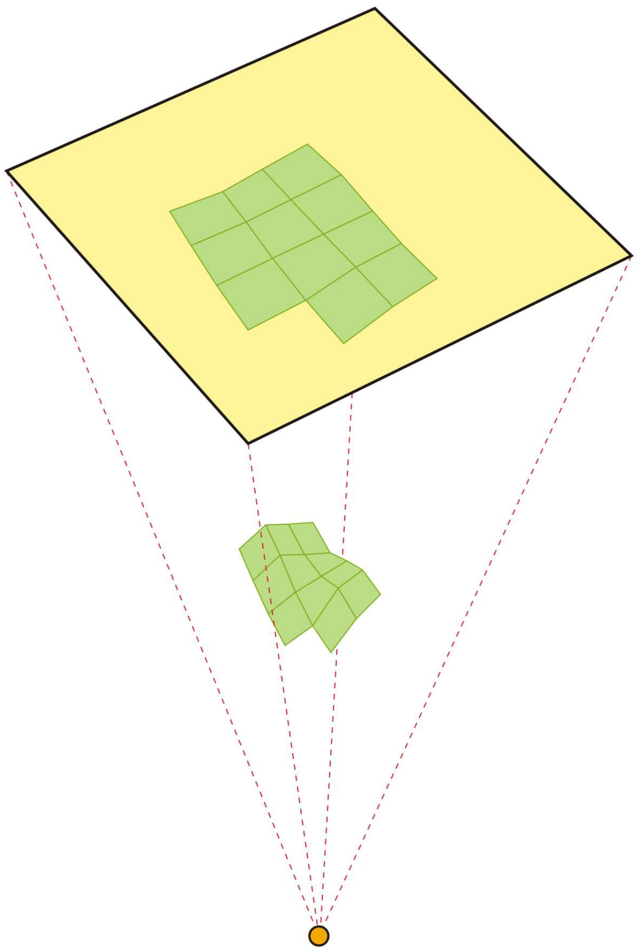


# Microquads

[Schwarz & Stamminger, EG 2007]

- Quads constructed from four adjacent unprojected samples that pass depth test





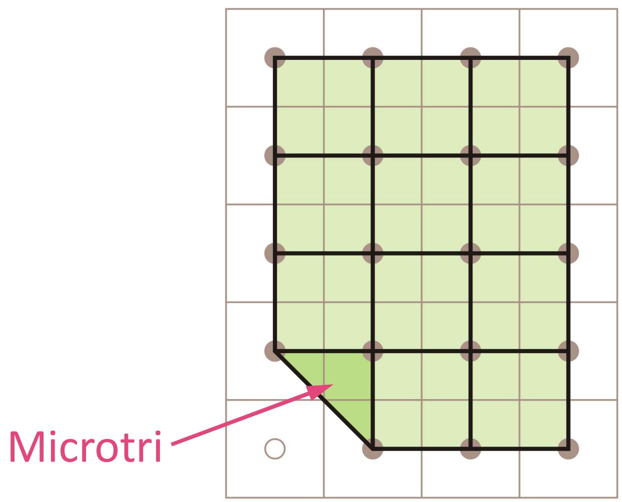
# Microquads

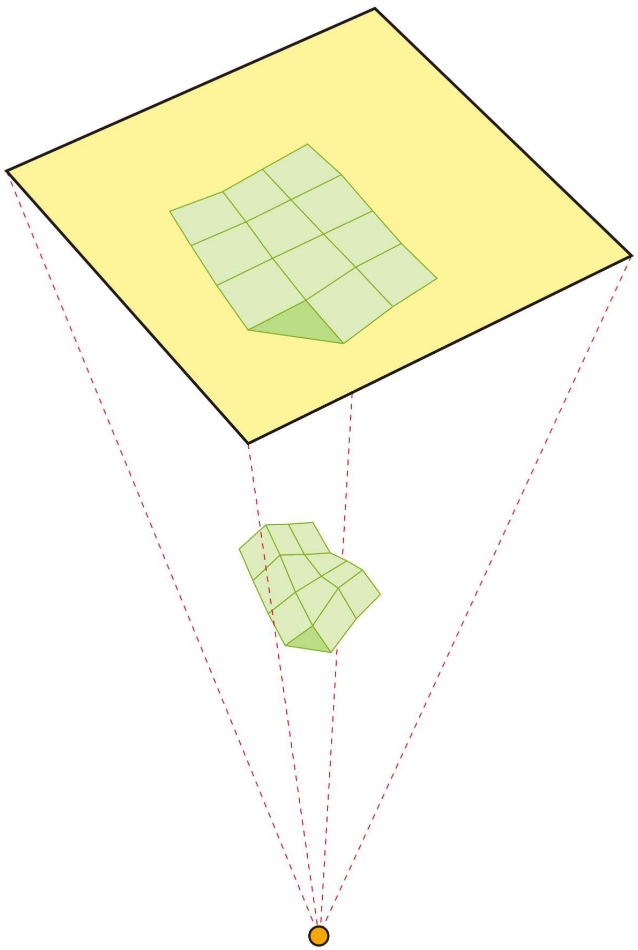
- No light leaks
- Piecewise-(bi)linear approximation
  - Often better fit to original geometry
  - Less prone to surface acne artifacts
- Tendency to underestimate occluders
  - May miss fine structures

# Microtris

[Schwarz & Stamminger, EG 2008 Short Paper]

- If only three out of 2×2 samples pass depth test, construct triangle from their unprojections

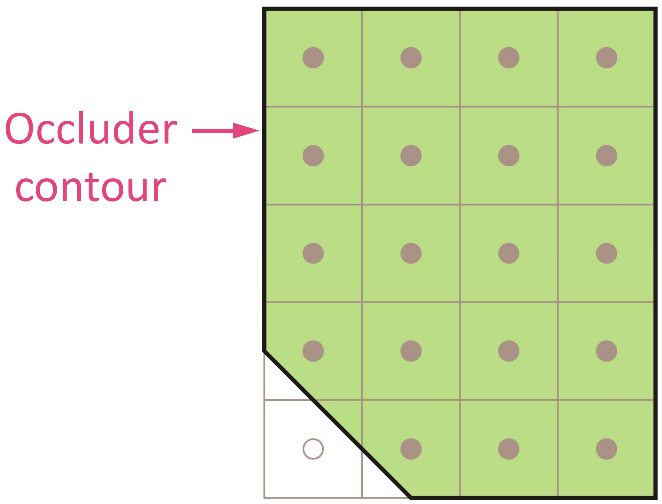


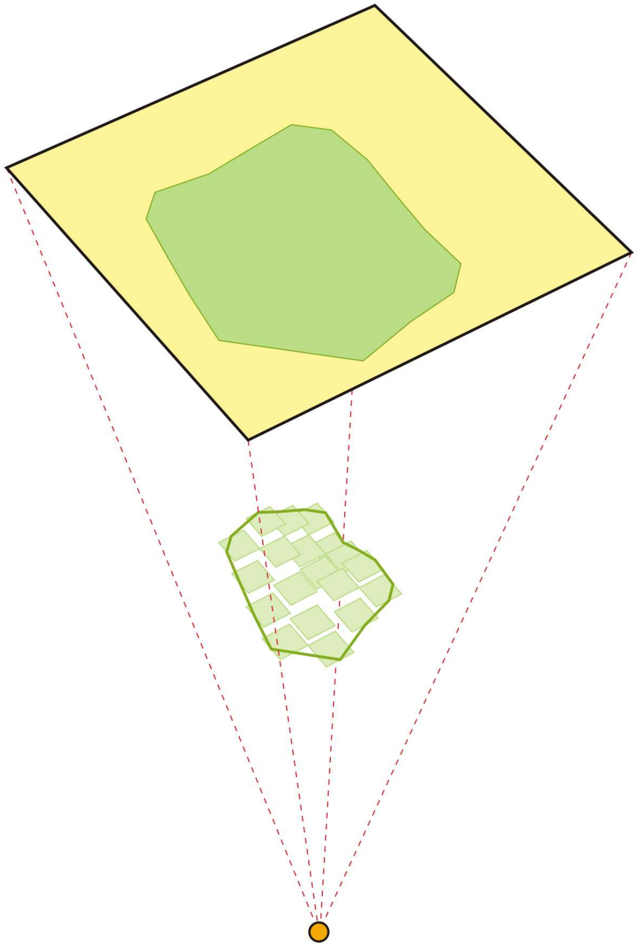



# Occluder Contours

[Guennebaud et al., EG 2007]

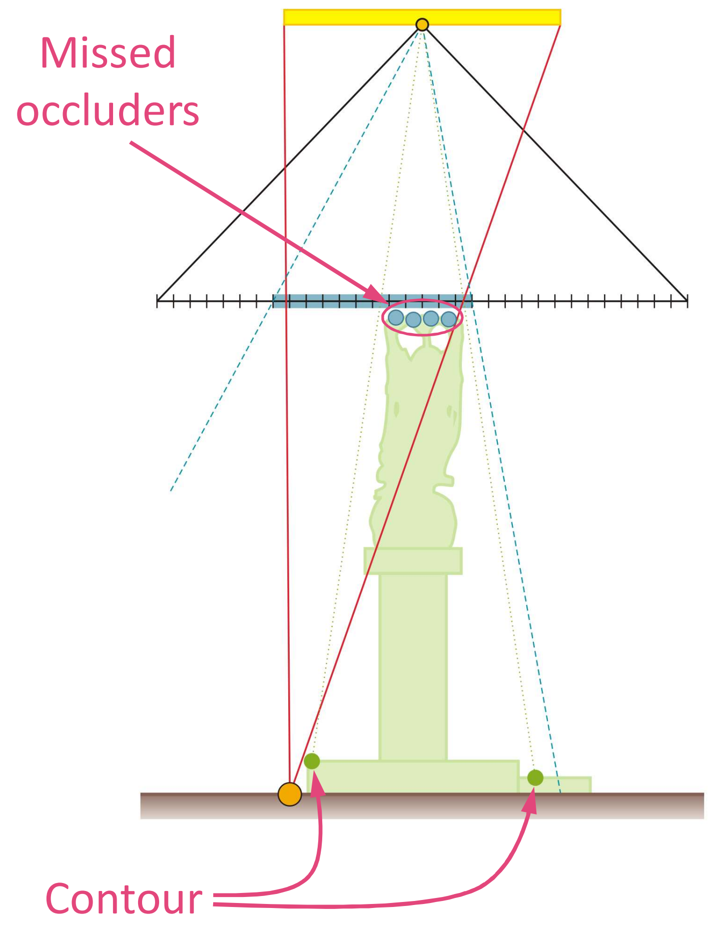
- Contours constructed for each connected region of samples passing depth test



# Occluder Contours

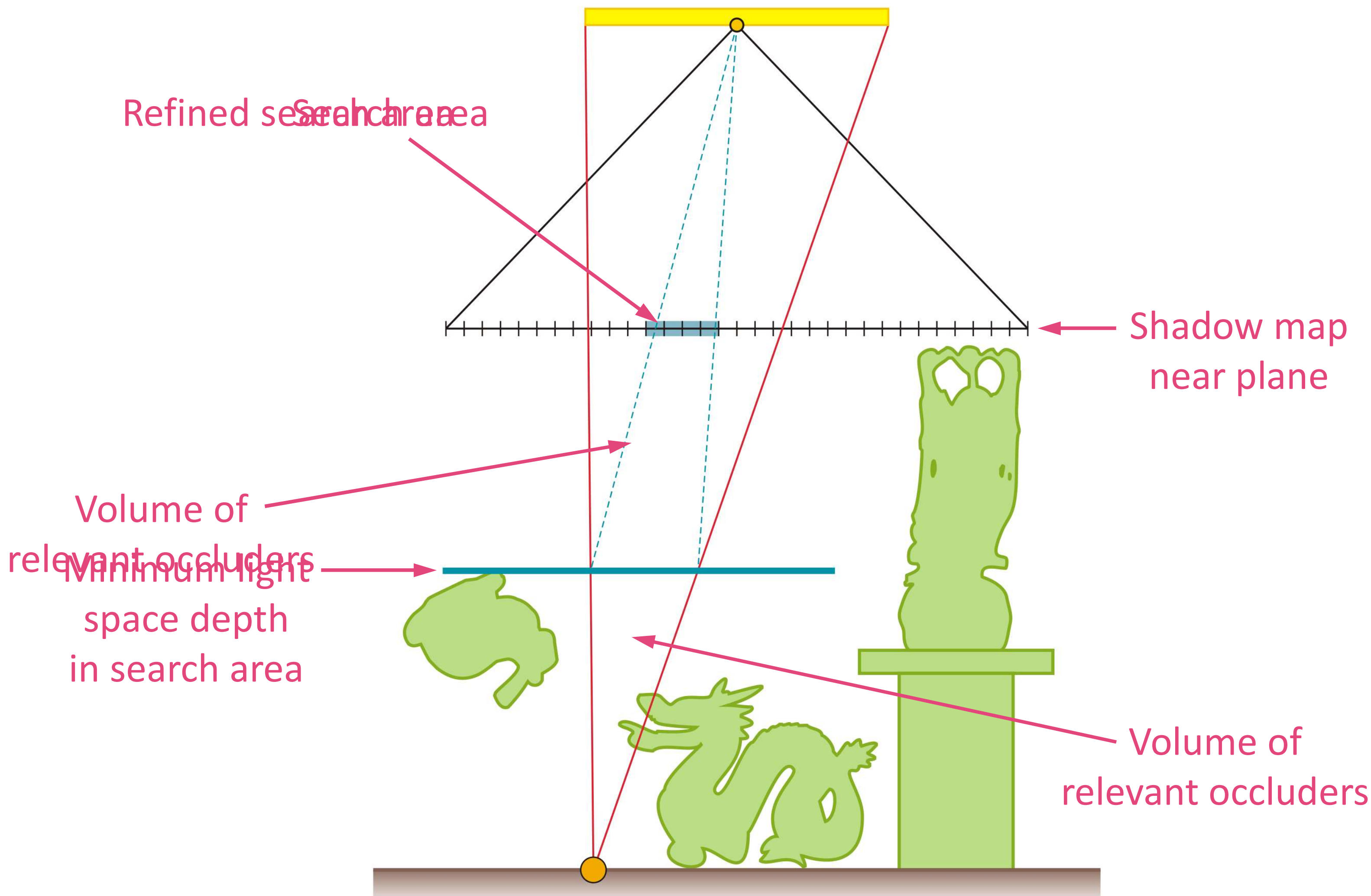
- Basically no light leaks
- Area determination via radial area integration
- Contour ≠ Silhouette
  - Occluders recorded in shadow map may be missed
- But also causes coherence
  - Neighboring receiver points often exclusively process identical contours
  - Allows for packet-based approach [Yang et al., EGSR 2009]



# Backprojection Acceleration

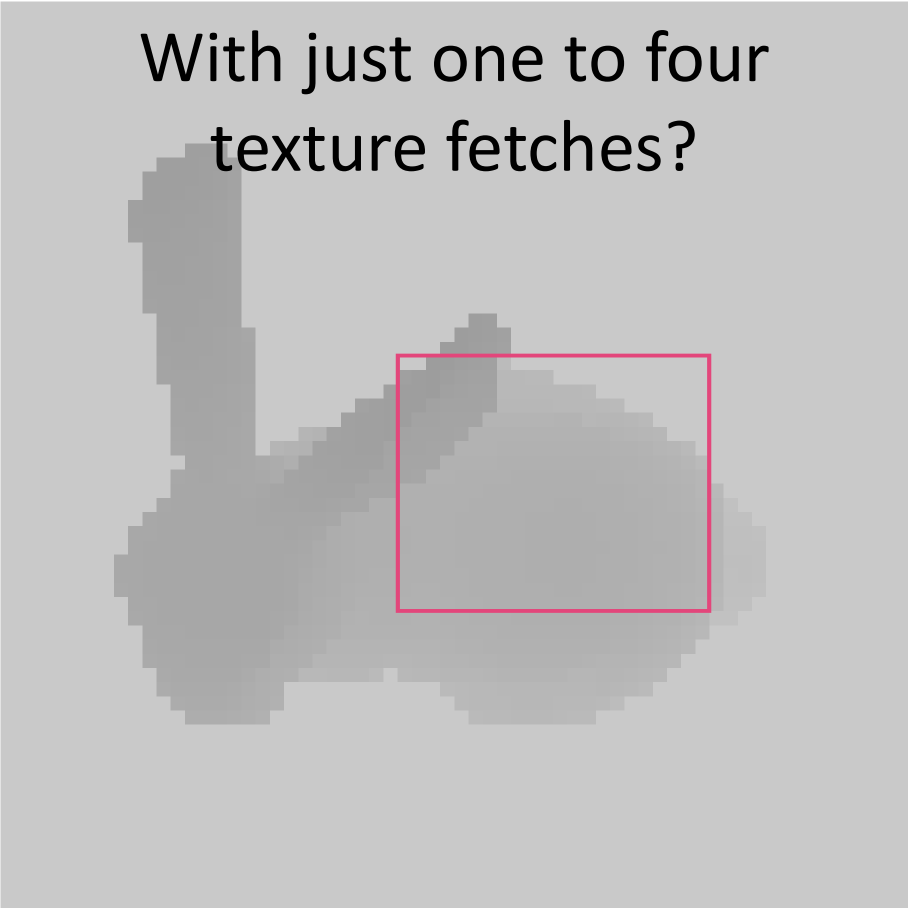
- Avoid useless computations
  - Search area pruning
  - Multi-scale representations
  - Hierarchical occluder construction
- Adapt accuracy
  - Micro-occluder subsampling
  - Coarser occluder approximations
  - Subsampling in screen space

# Search Area Determination



# Acceleration Structures

How to determine depth range of a shadow map region?



# Hierarchical Shadow Map

[Guennebaud et al., EG 2006]

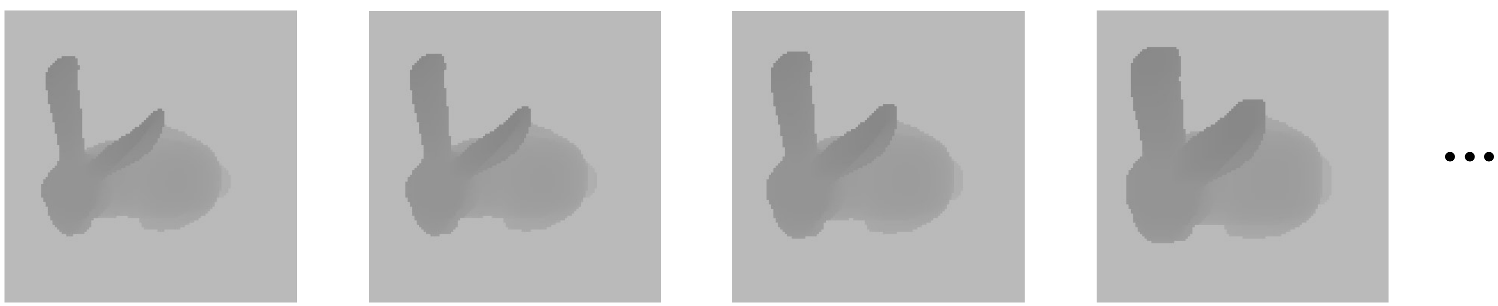
- Min/max pyramid of shadow map (hierarchical z-buffer)
- Stored in mipmap chain of shadow map

- Cheap, but often yields loose search areas
- Why not SAT: cannot express min/max

# Multi-scale Shadow Map

[Schwarz & Stamminger, EG 2007]

- Stack of depth ranges for all power-of-two-sized neighborhoods
- Stored in array texture

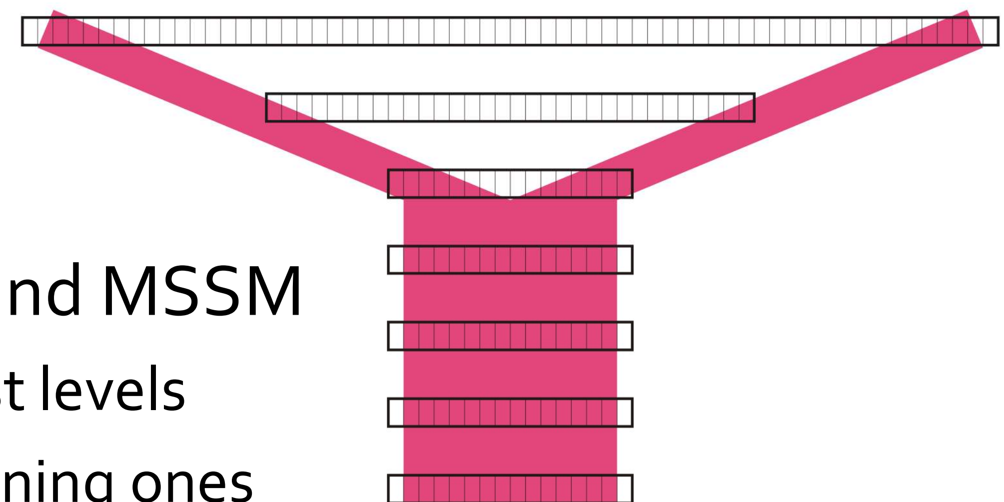


- Tight search areas, but can be costly
  - Cache efficiency (no small textures)
  - Generation

# Hybrid Y Shadow Map

[Schwarz & Stamminger, GI 2008]

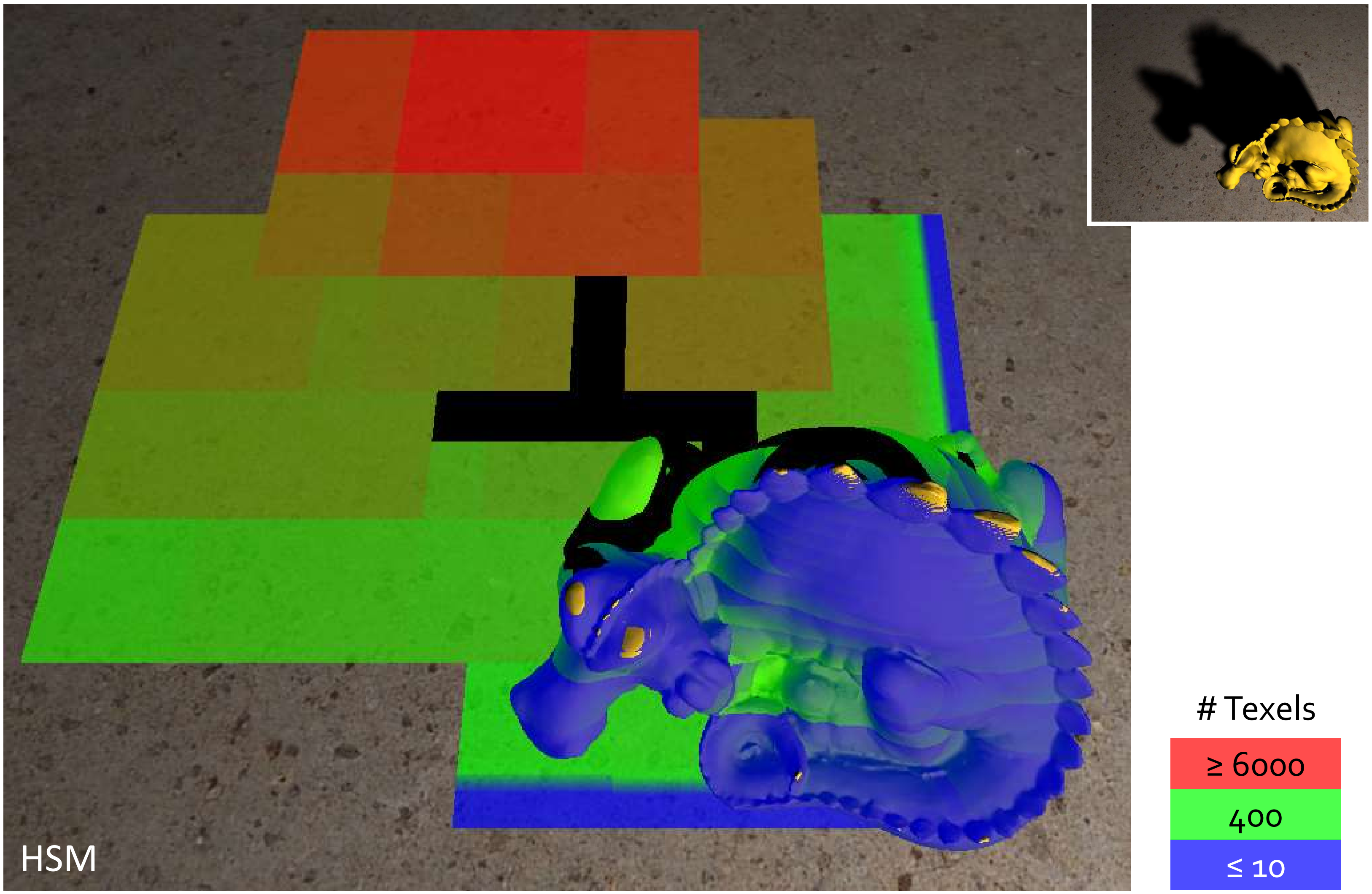
- Goal: Get the best of both
  - Low cost from HSM
  - Tight search areas from MSSM



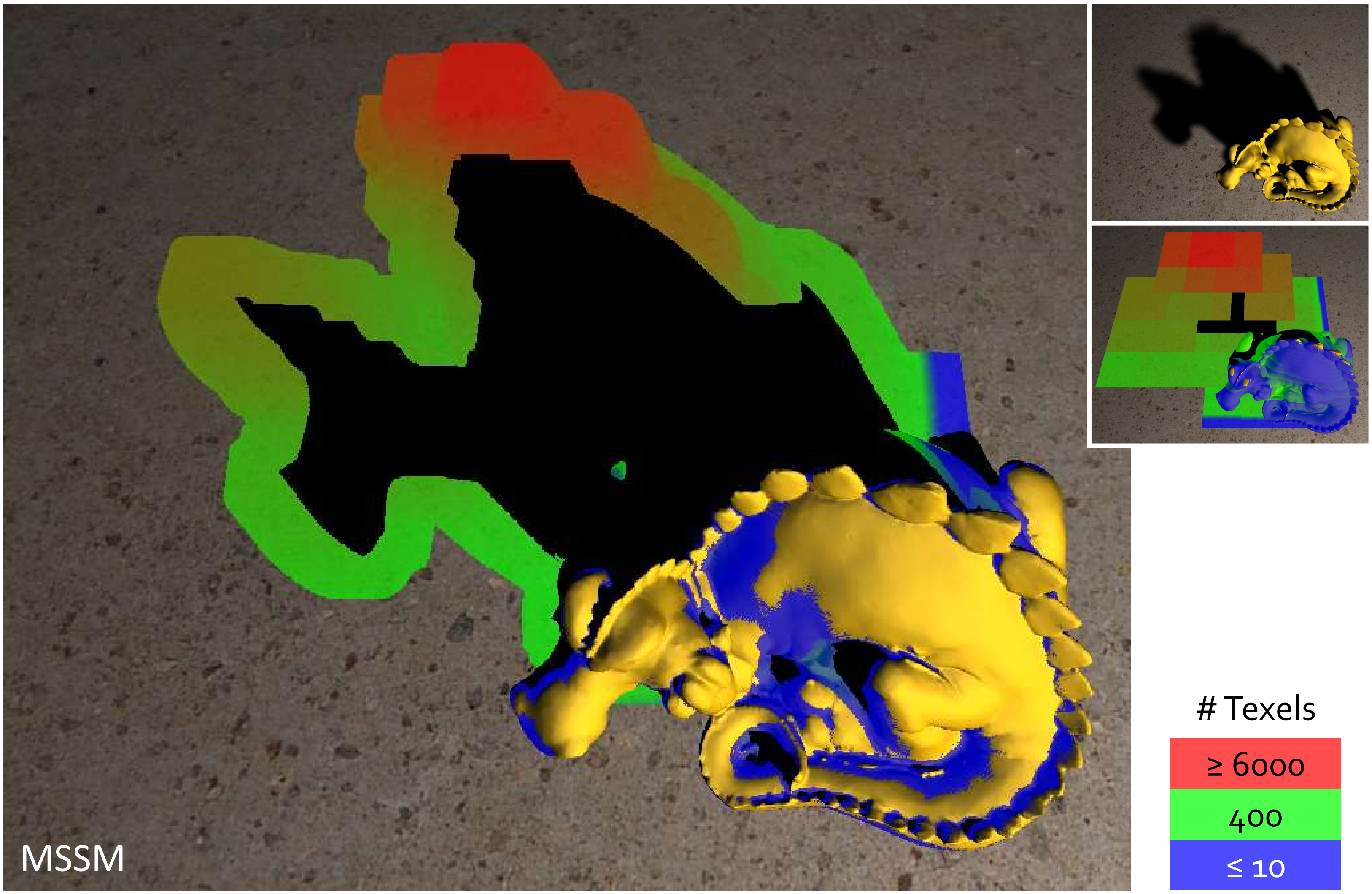
- Hybrid between HSM and MSSM
  - Pyramid (HSM) for finest levels
  - Stack (MSSM) for remaining ones

# Acceleration Structures

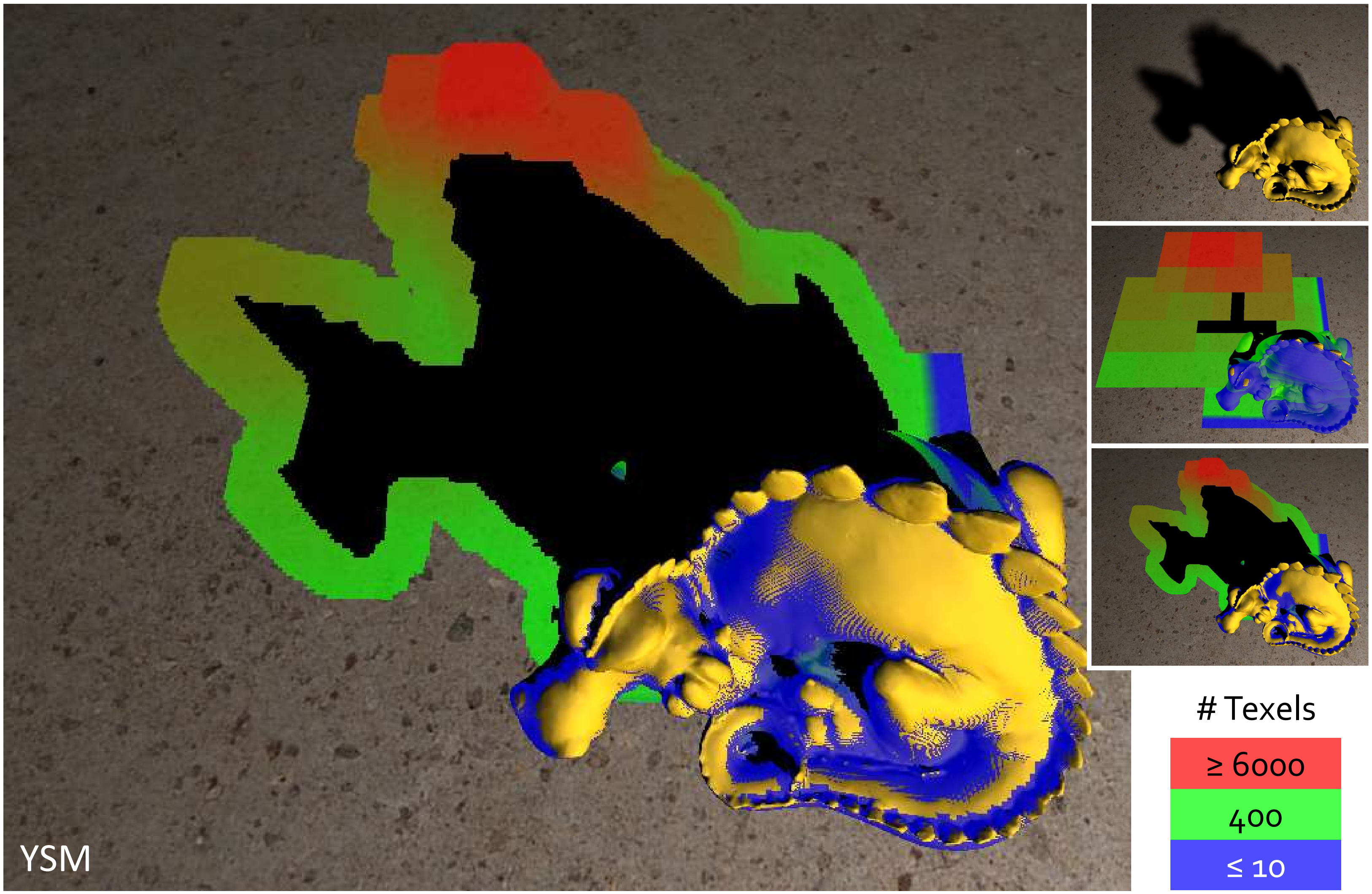
# Acceleration Structures



# Acceleration Structures



# Acceleration Structures



# Hierarchical Occluder Construction

- HSM is a quadtree constructed over shadow map
  - Traverse this tree to identify and process relevant micropatches

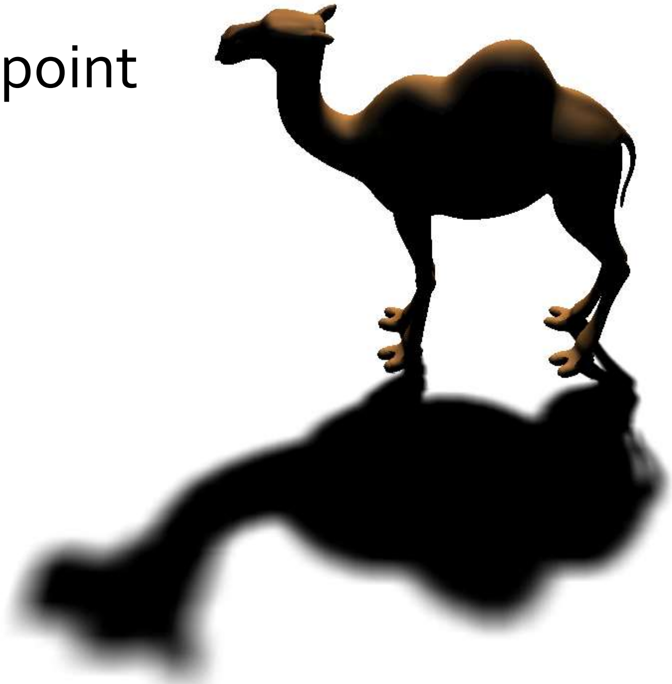
    [Dmitriev, 2007]

- MSSM is essentially a forest of quadtrees
  - Hierarchically extract occluder contours

    [Yang et al., EGSR 2009]

# Coarser Occluder Approximations

- Goal: Limit number of processed shadow map texels
- Approach: Use coarser-resolution shadow map
  - Can take appropriate HSM level
  - Level selection is done per receiver point
  - Correct depth bias?
- May lower visual quality
  - Decreased smoothness and detail in the shadow shape
  - Changes in shadow region size



# Coarser Occluder Approximations

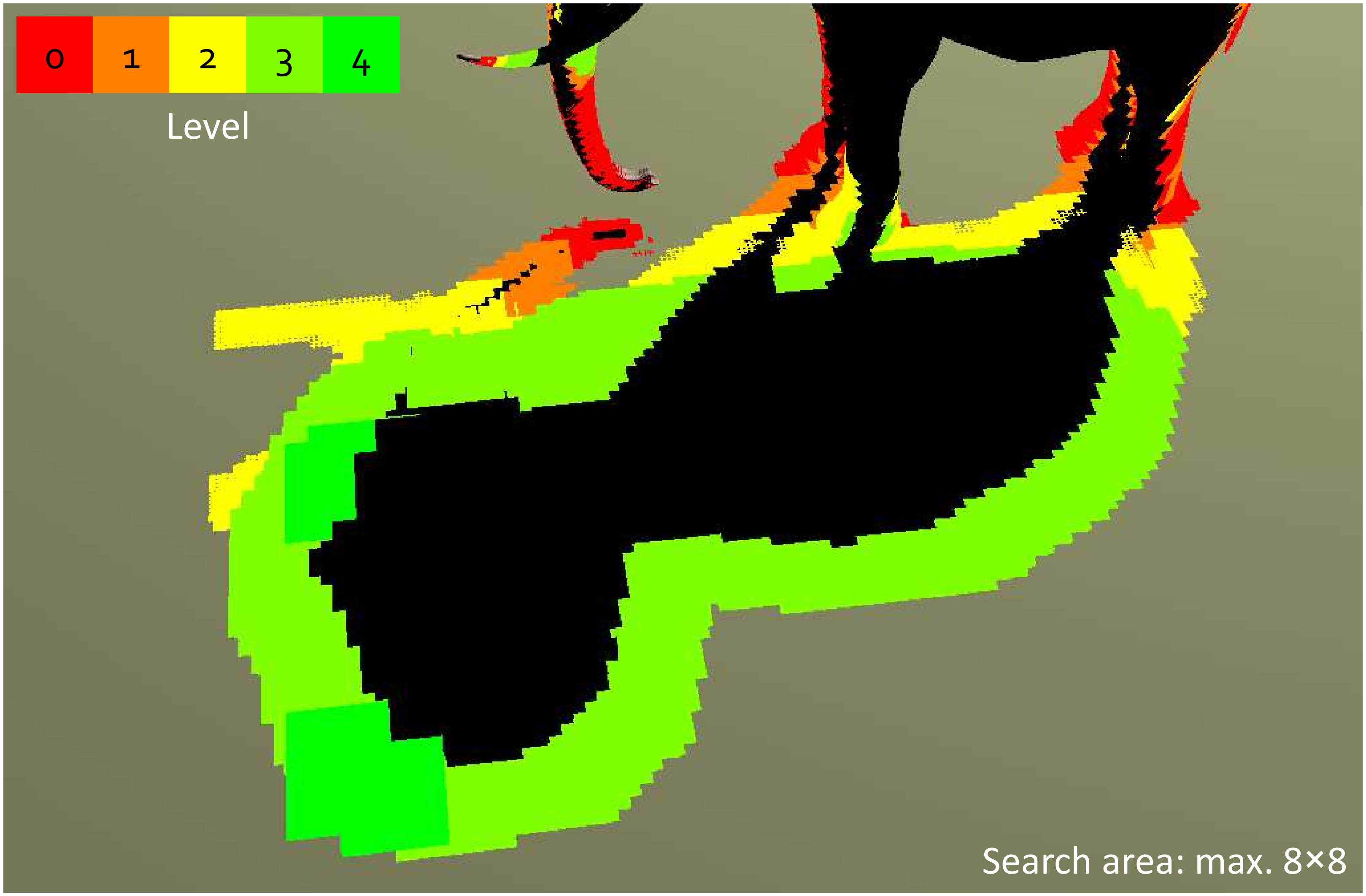
# Coarser Occluder Approximations

# Coarser Occluder Approximations

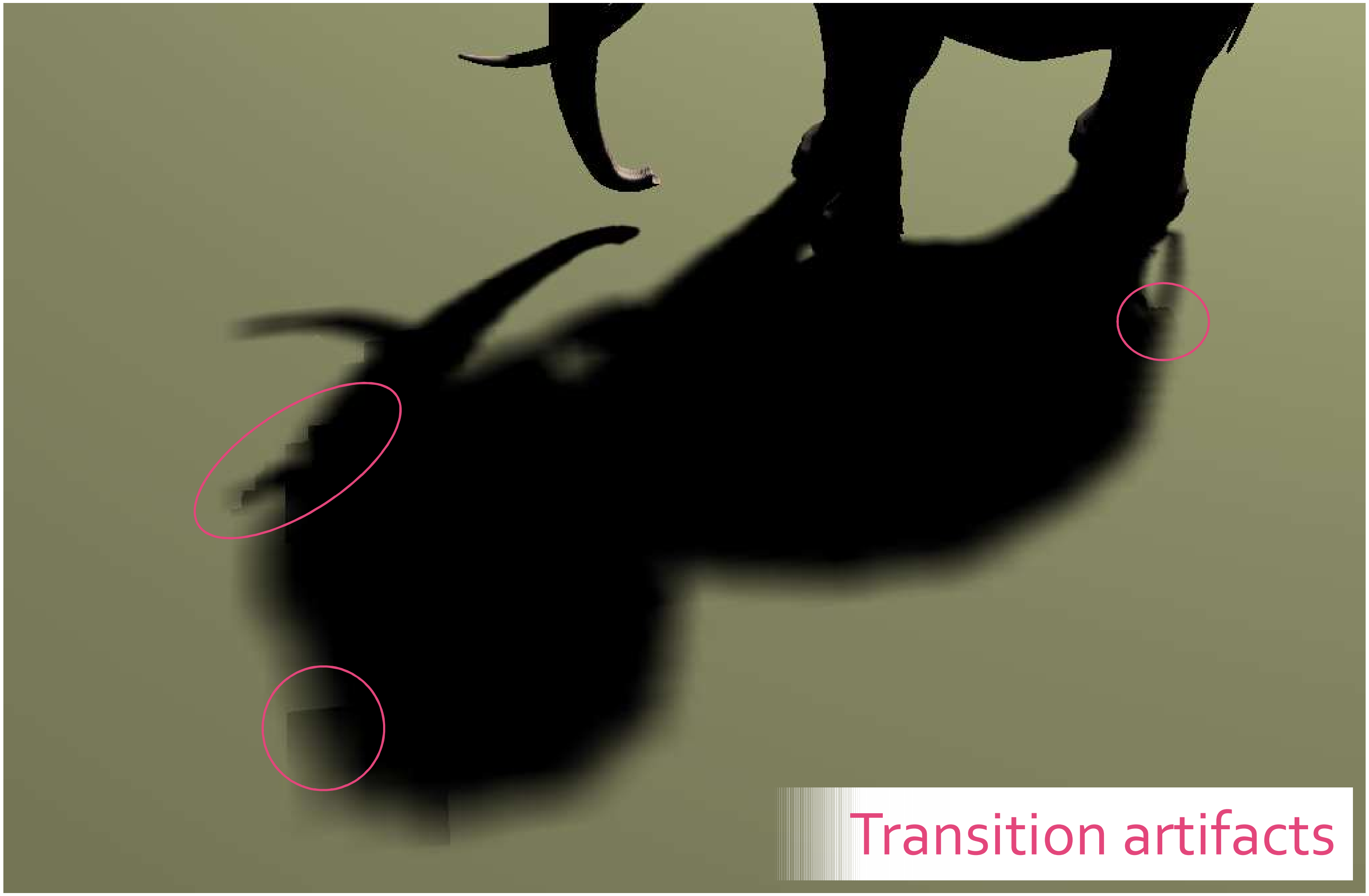
# Coarser Occluder Approximations

# Coarser Occluder Approximations

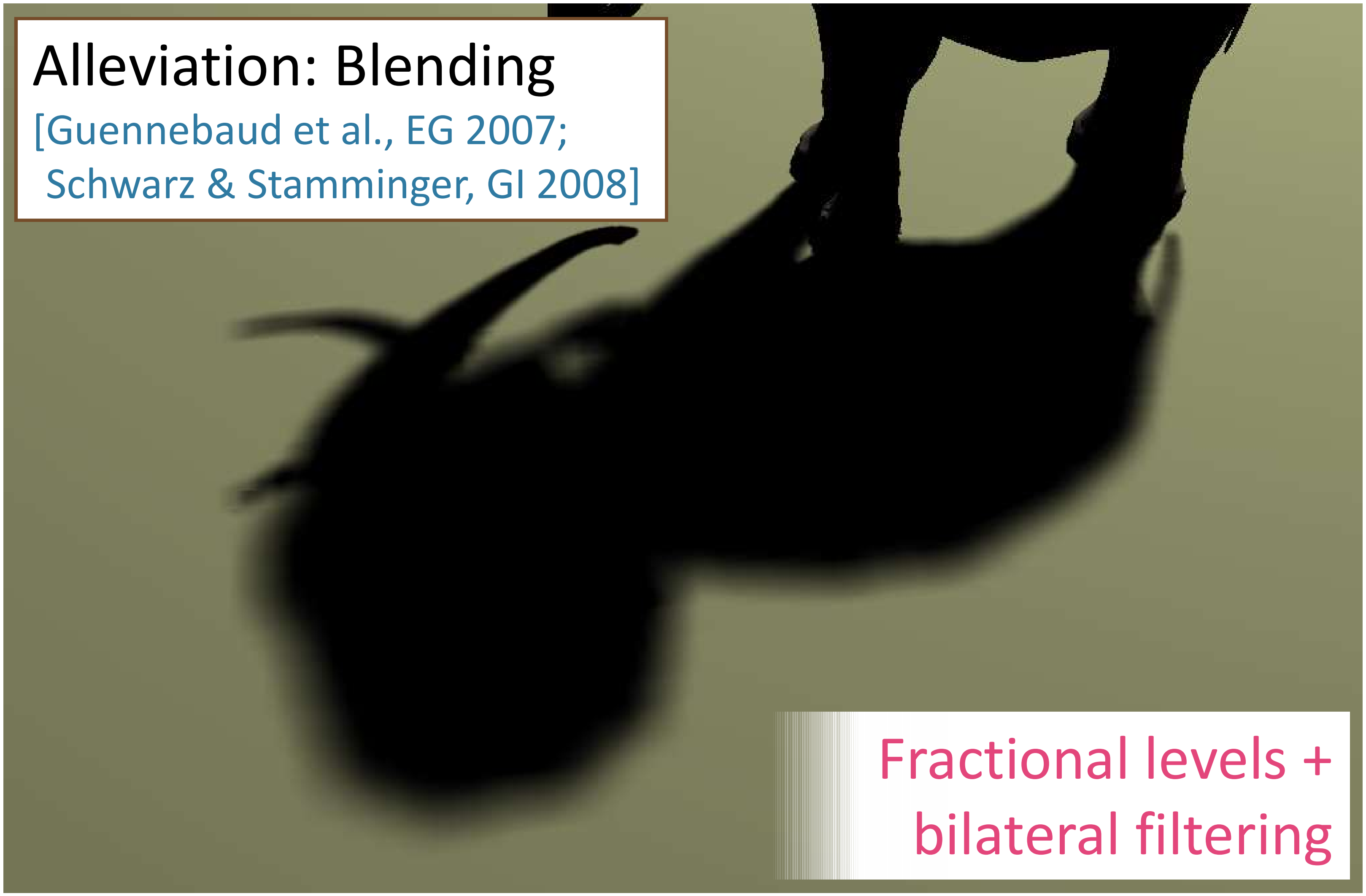
# Coarser Occluder Approximations



# Coarser Occluder Approximations



# Coarser Occluder Approximations

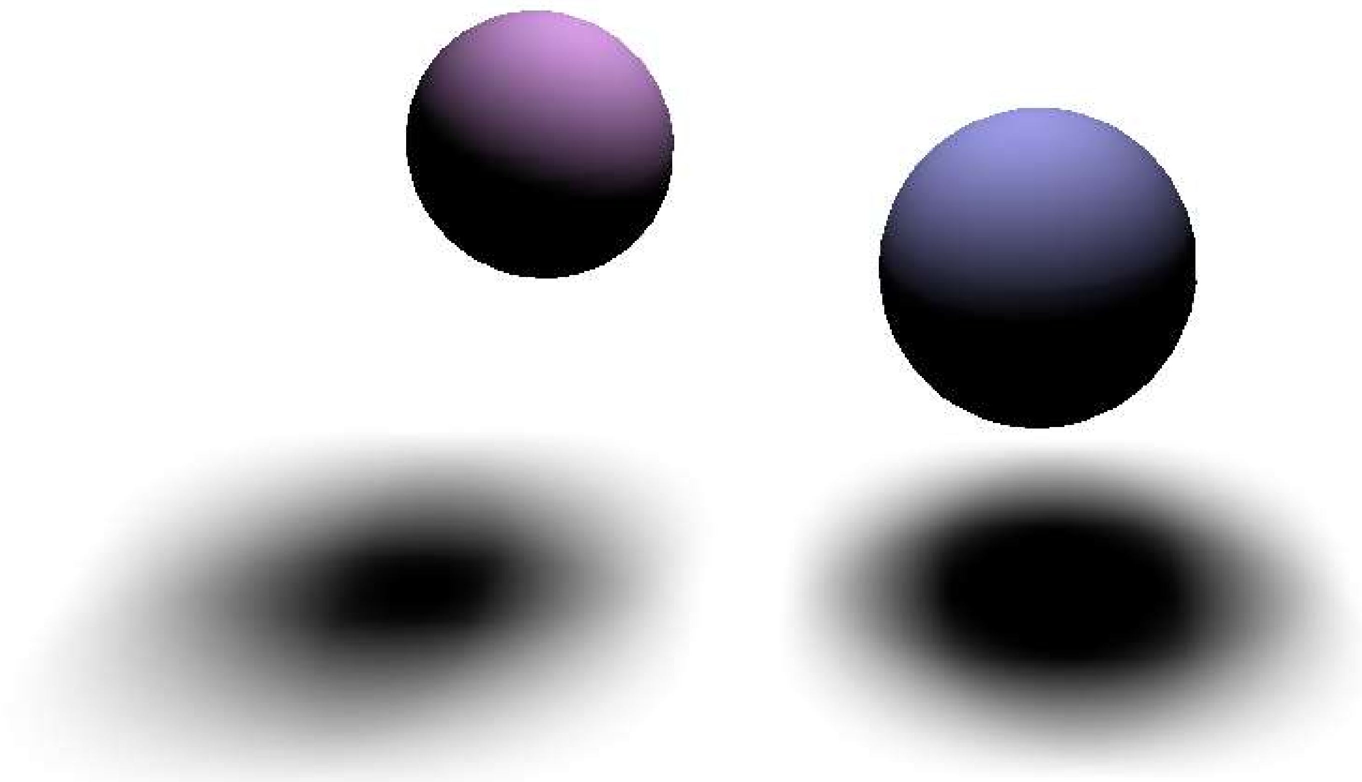


**Alleviation: Blending**
[Guennebaud et al., EG 2007;
Schwarz & Stamminger, GI 2008]

Fractional levels +
bilateral filtering

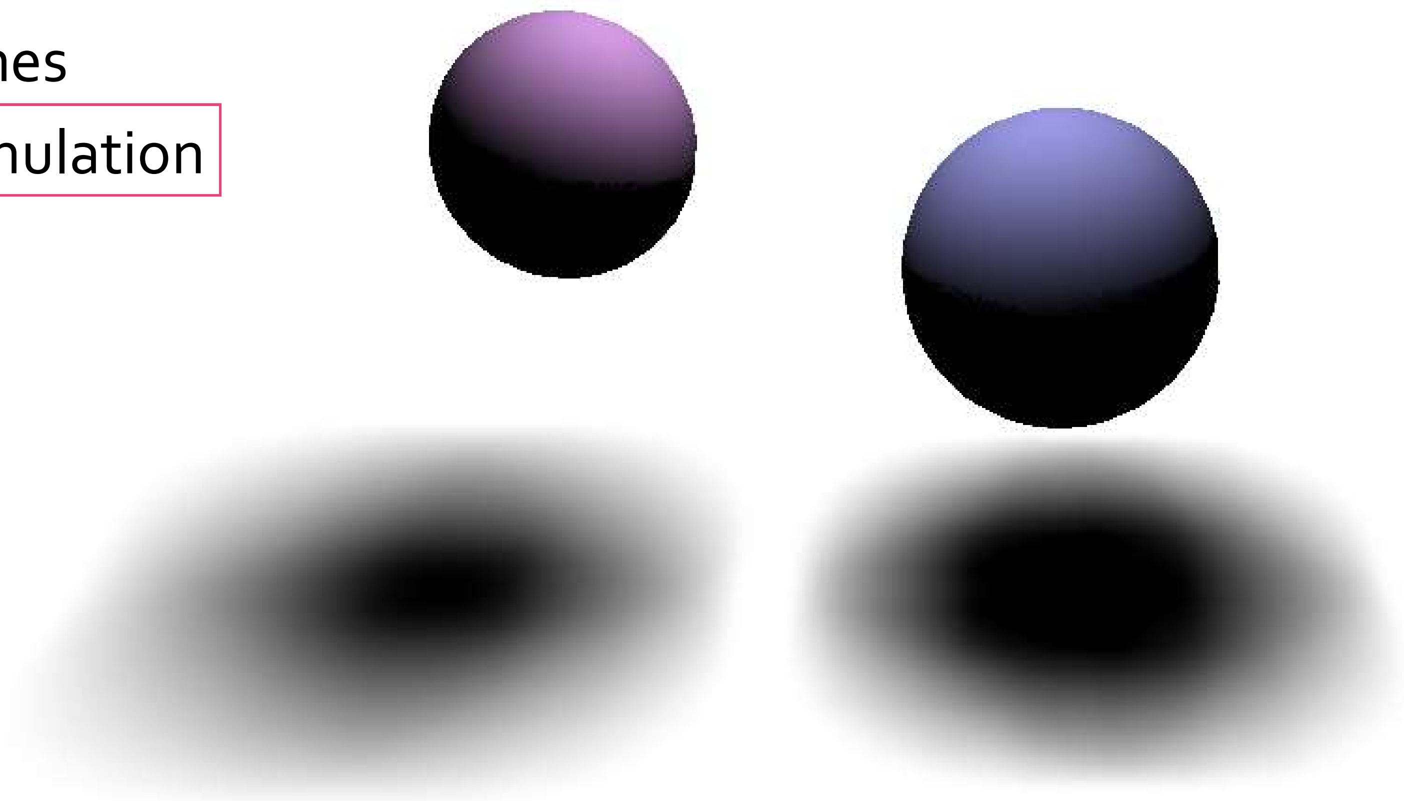
# Example

- Reference



# Example

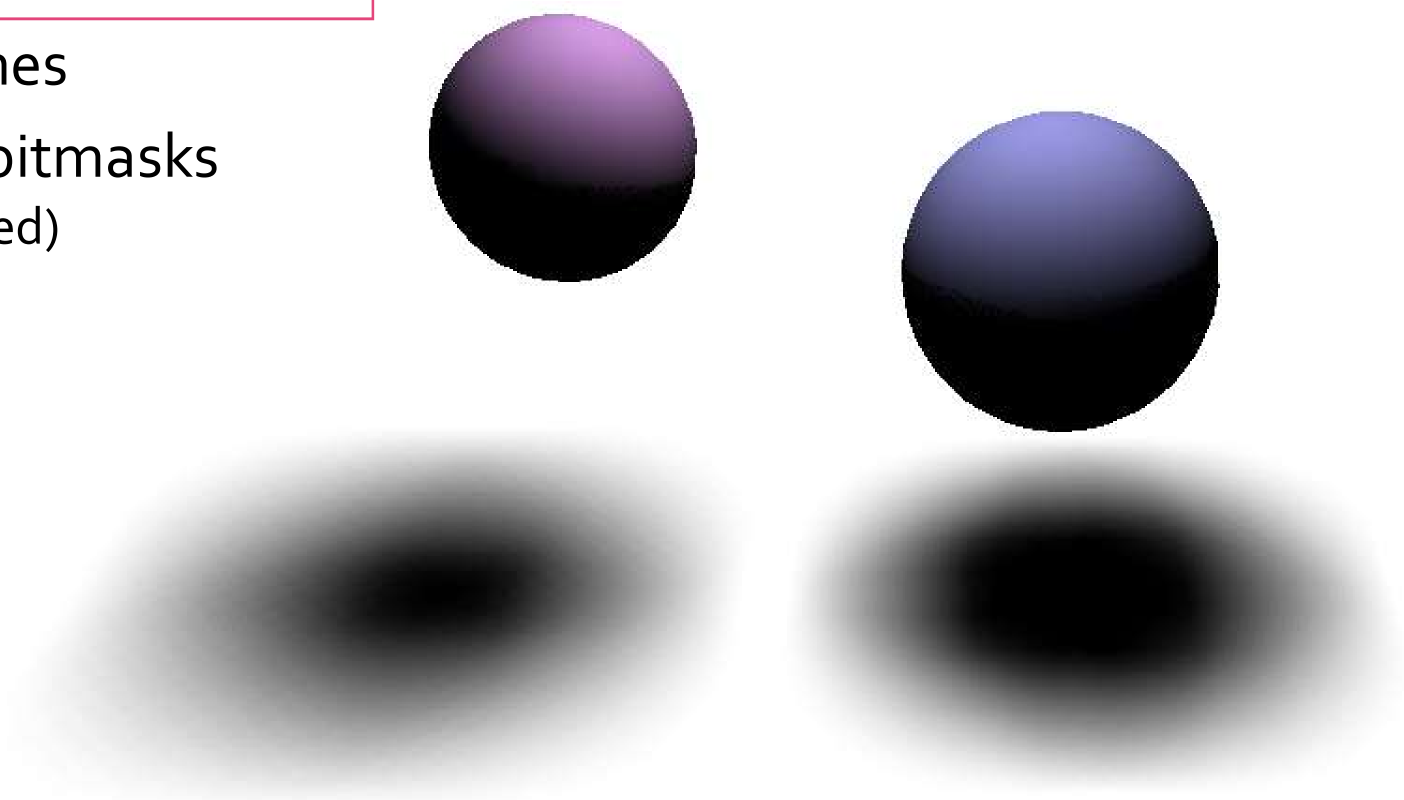
- Search area: max. 20×20
- Micropatches
- Area accumulation
- YSM



105 fps
(1024×1024, GeForce GTX 285)

# Example

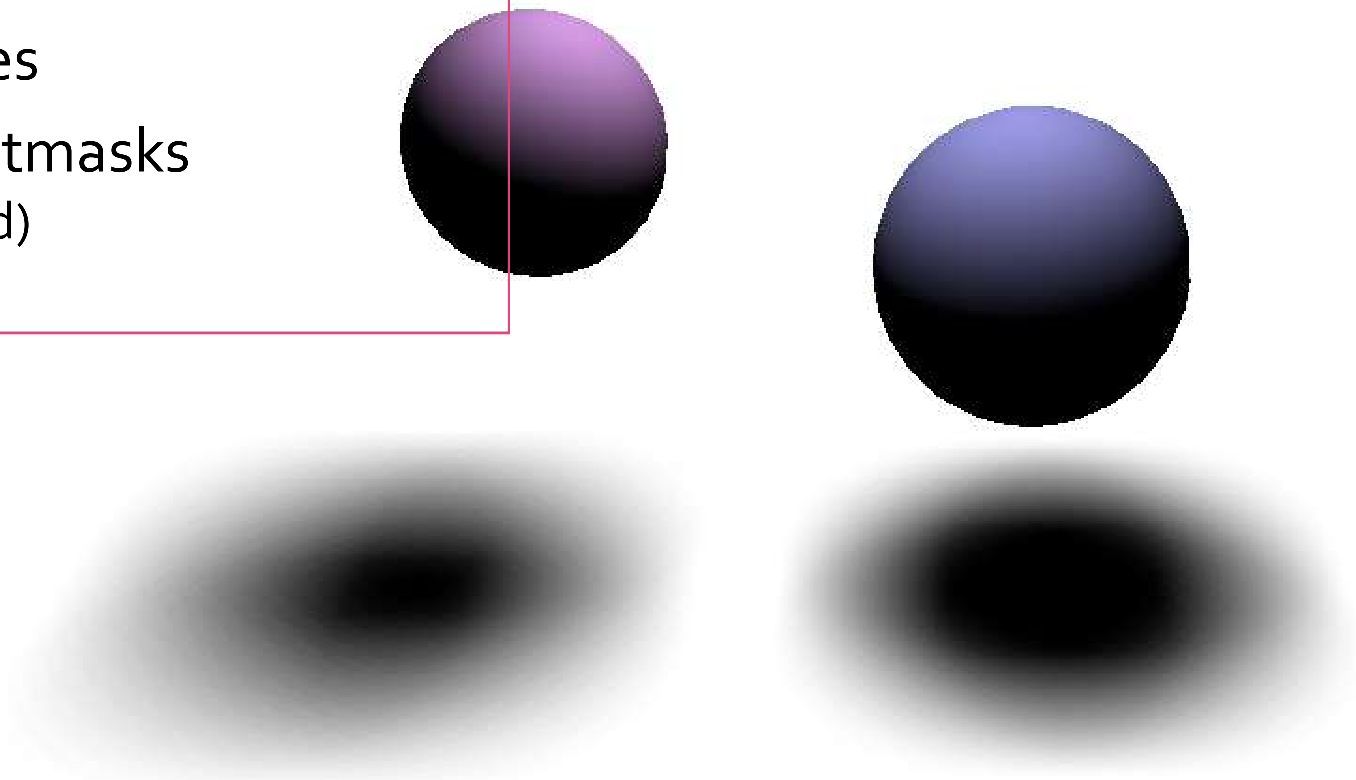
- $\boxed{\text{Search area: max. } 20\times20}$
- Micropatches
- Occlusion bitmasks
  (16×16, jittered)
- YSM



51 fps
(1024×1024, GeForce GTX 285)

# Example

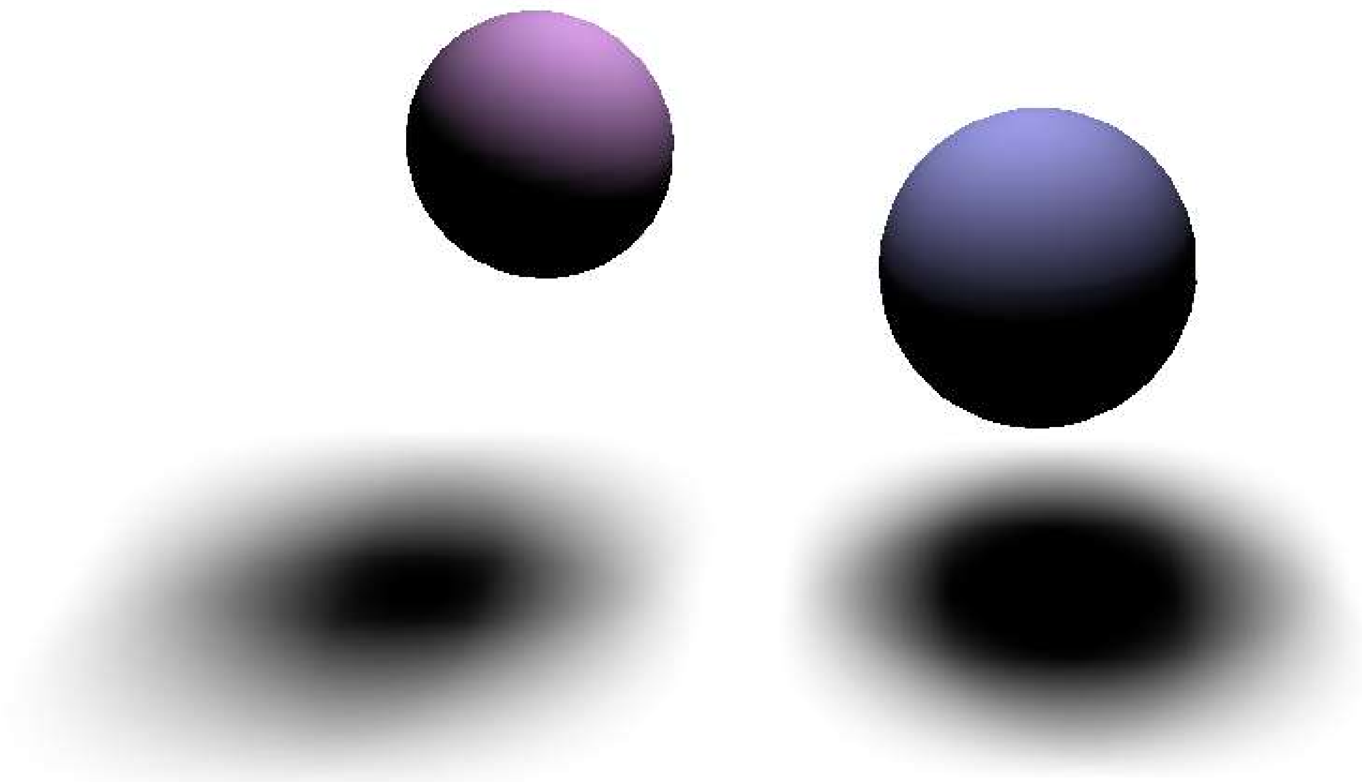
- No imposed search area bound
- Micropatches
- Occlusion bitmasks
  (16×16, jittered)
- YSM



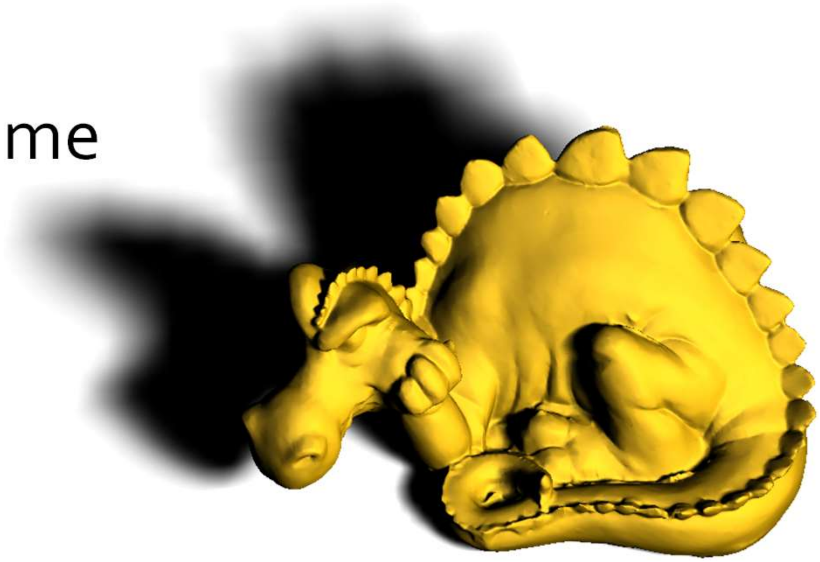
$< 1$ fps
(1024×1024, GeForce GTX 285)

# Example

- Reference again



# Soft Shadow Mapping



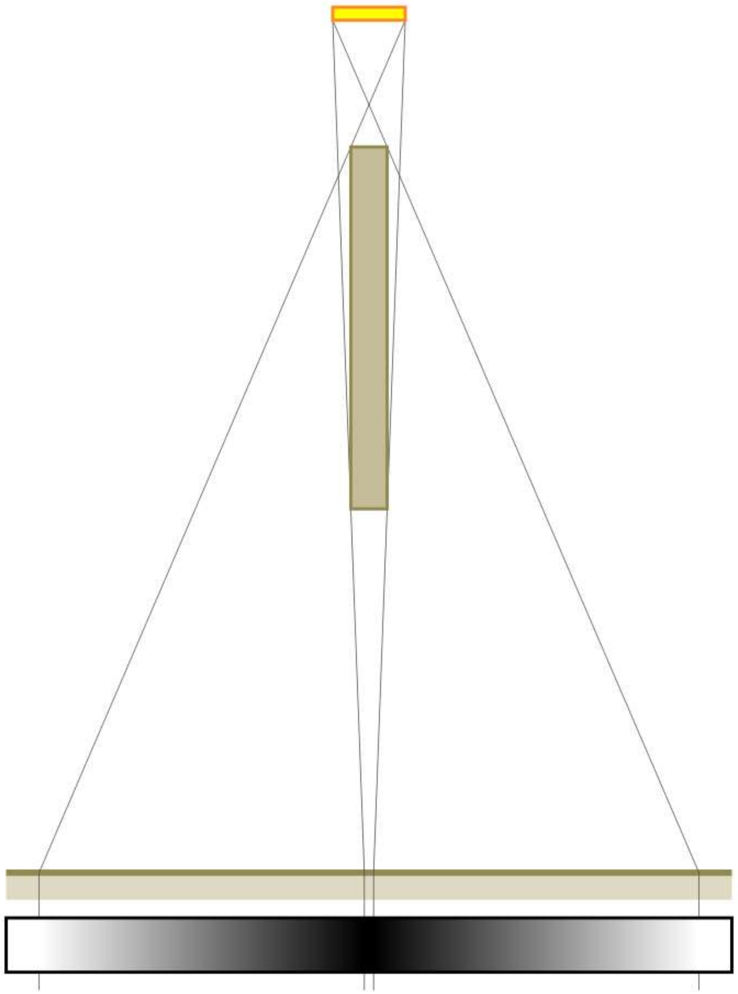
+ Physically plausible
+ Rather high quality at real-time frame rates possible

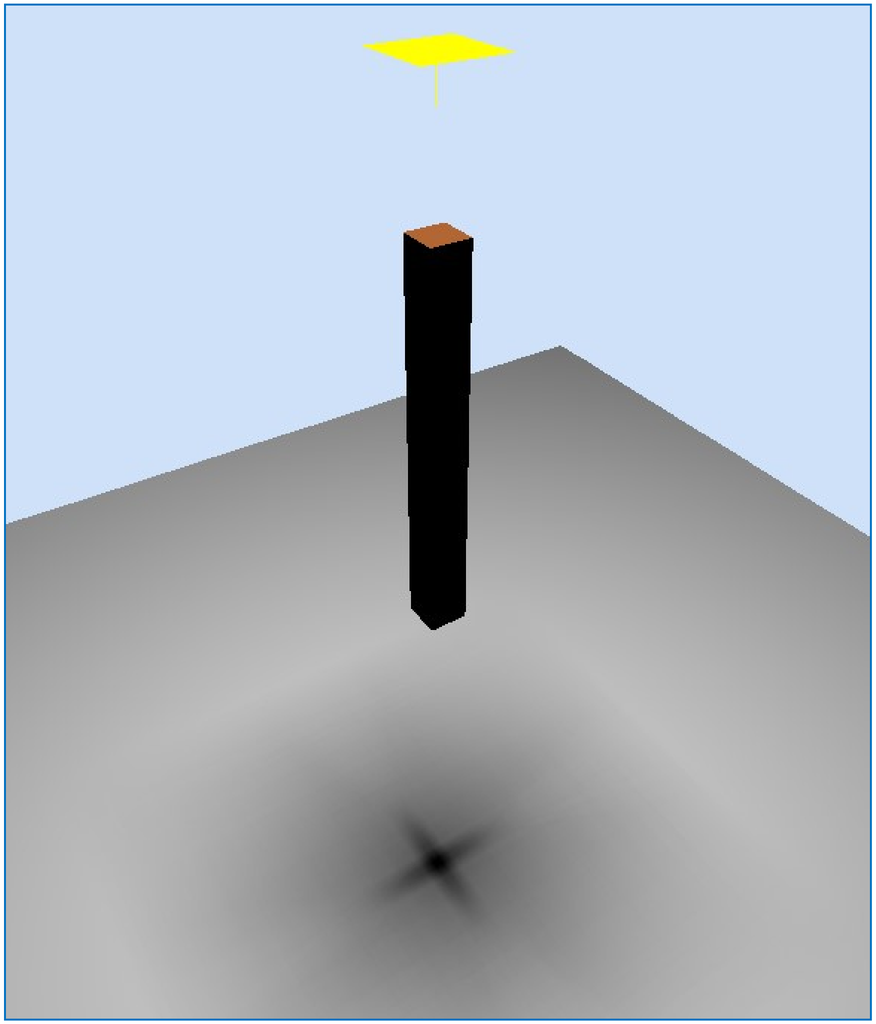
- Performance strongly dependent on
  - Search area size
  - Number of pixels requiring back projection

– Uses only approximation of subset of occluders
  - Typically those visible from the light source's center

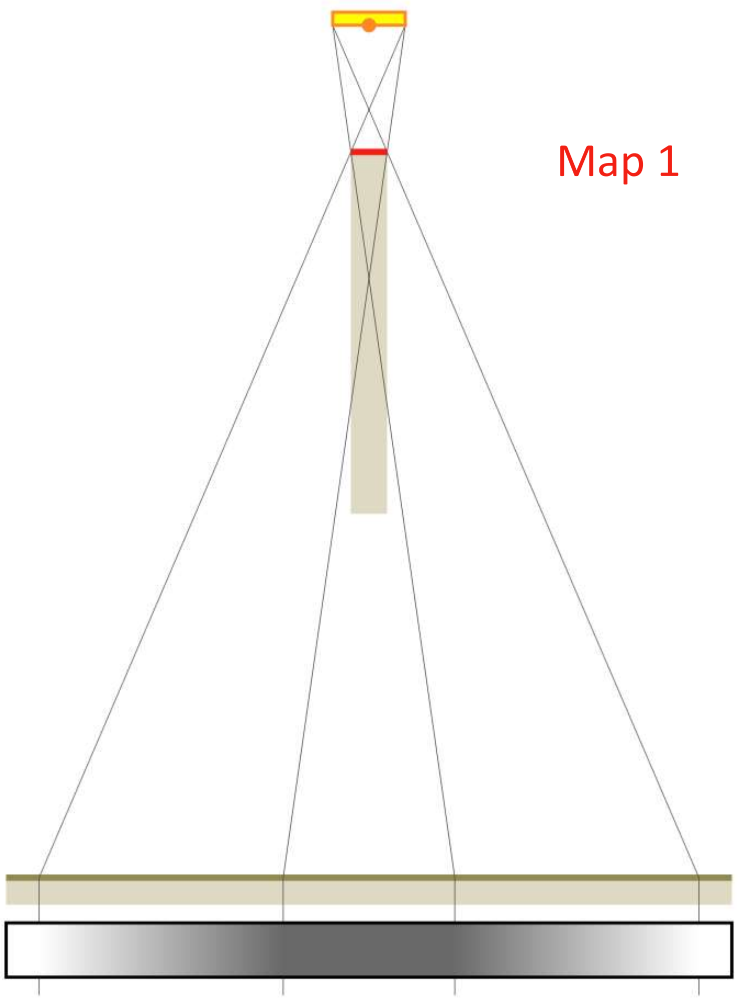
– Usually all gaps are closed invariably

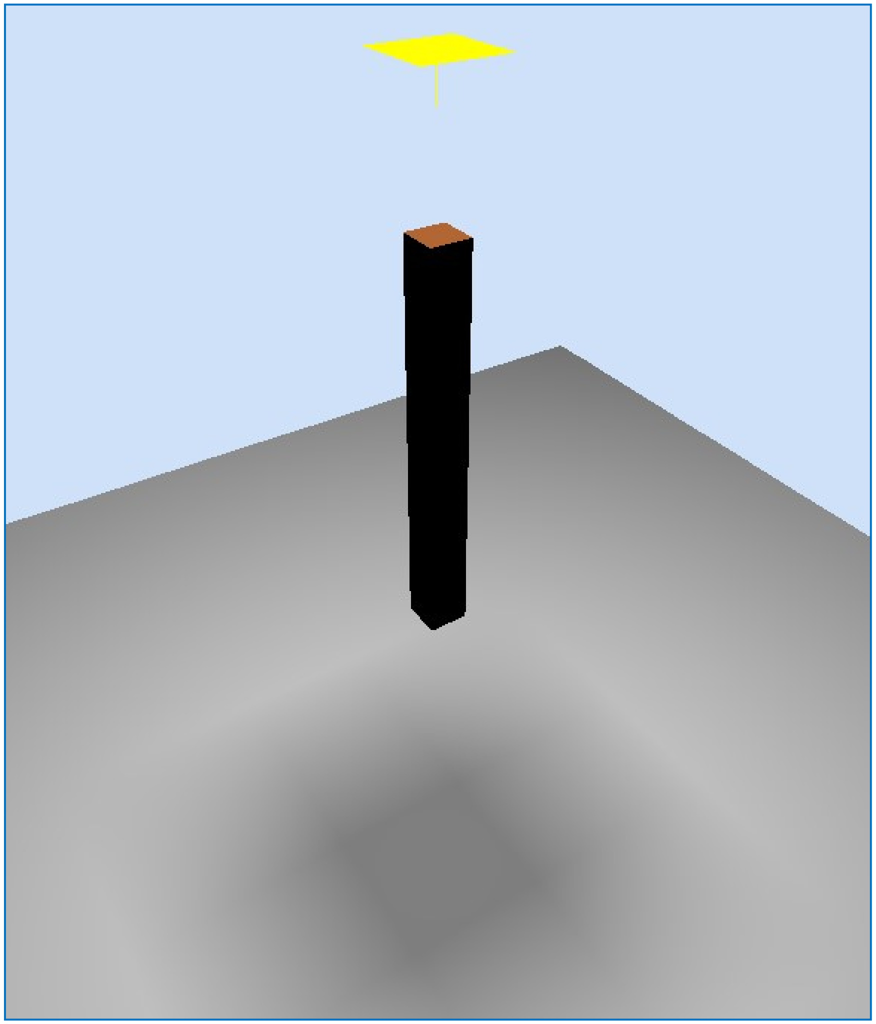
# Multiple Shadow Maps



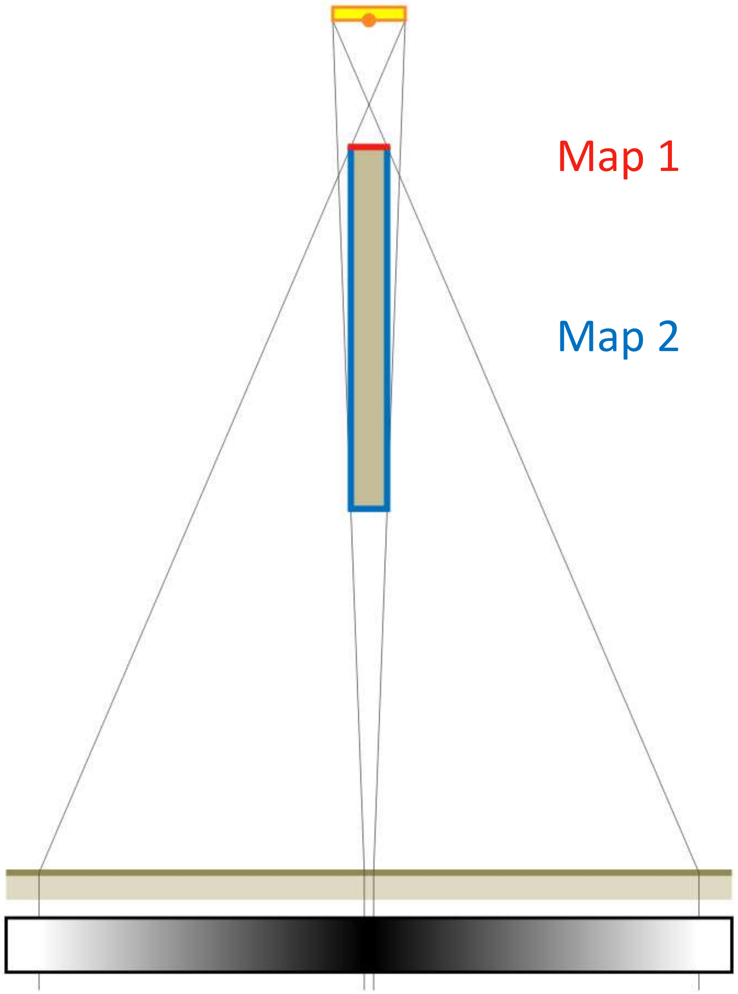


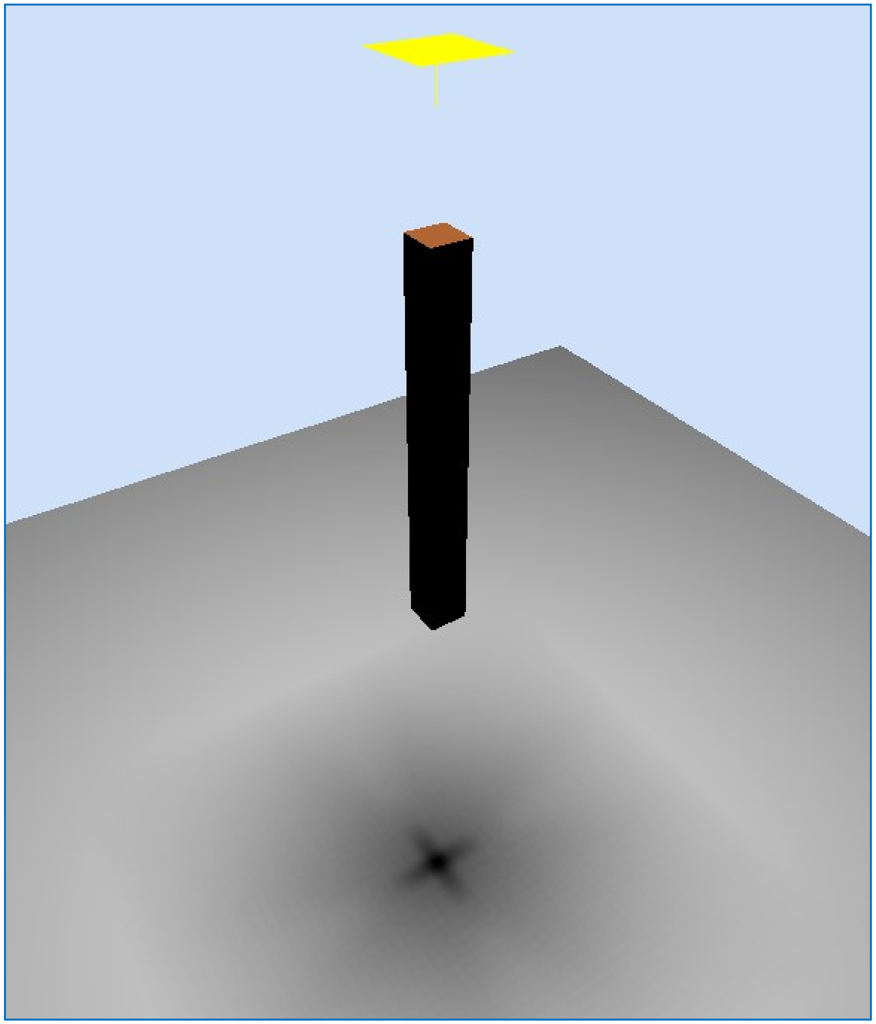
# Multiple Shadow Maps



# Multiple Shadow Maps



# Multiple Shadow Maps

- Acquire shadow maps from several points on light
- Merge them into an extended shadow map

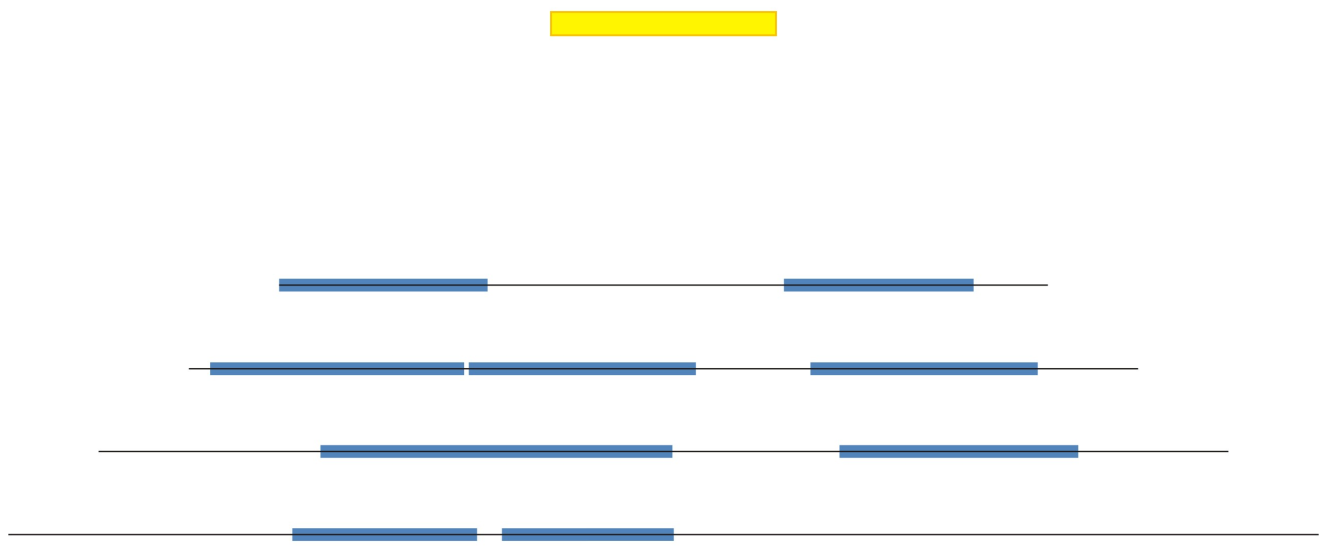
- Layered attenuation maps
  [Agrawala et al., SIGGRAPH 2000]
- Penumbra deep shadow maps
  [St-Amour et al., GI 2005]

- Raytracing against multi-layered shadow maps
  [Lischinski & Rappoport, EGWR 1998; Keating & Max, EGWR 1999; Agrawala et al., SIGGRAPH 2000; Xie et al., EGSR 2007]
- Merging by exploiting temporal coherence
  [Scherzer et al., ISVC 2009]

# Occlusion Textures

[Eisemann and Décoret, SIBGRAPI 2006]

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



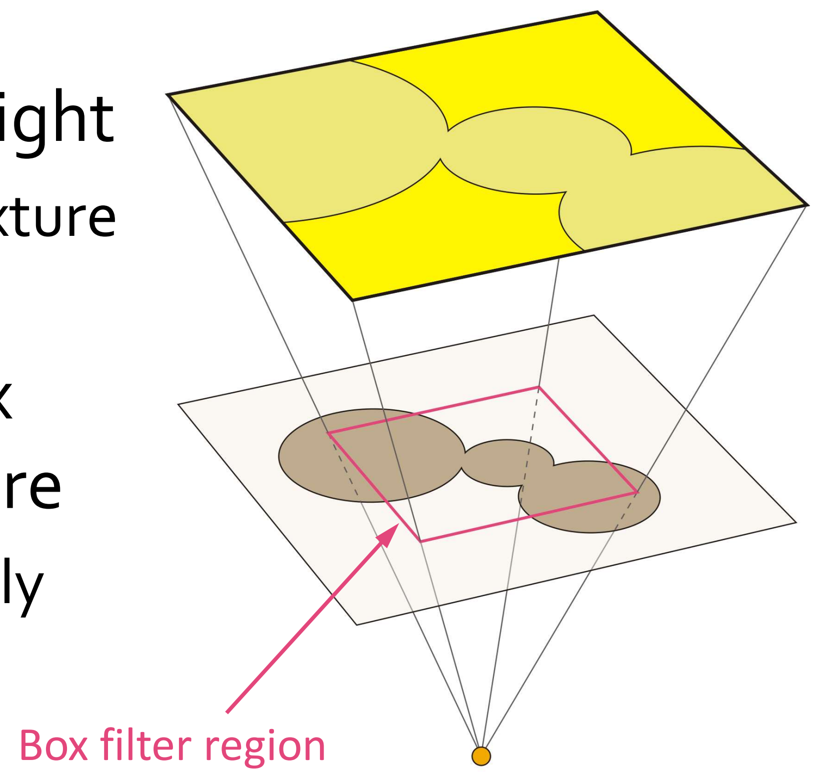
# Occlusion Textures

- The covered parts of each slice are encoded in a binary occlusion texture

# Visibility Computation = Filtering

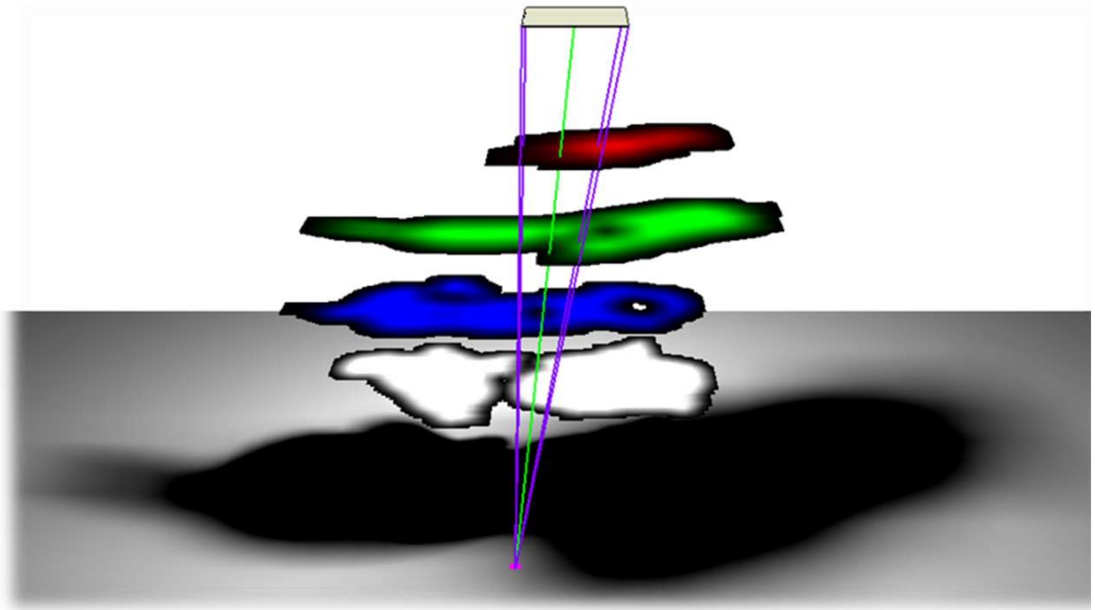
[Soler and Sillion, SIGGRAPH 1998]

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



# Occlusion Textures

- Pre-filter occlusion textures
  - Mip-mapping
  - N-buffers
  - Summed-area table



- For each blocking slice, lookup appropriately filtered response in pre-filtered occlusion texture
  - Accumulate shadow contributions (multiplicatively)

# Occlusion Textures

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

- Mainly suited for compact indoor environments

– Heuristic occluder fusion handling
– Discretization of scene into small number of slices can cause some quality problems

# Image-based Approaches

- Percentage-closer soft shadows
- Soft shadow mapping
- Occlusion textures

- Use sampled representation of (subset of) occluders
  - Supports versatile geometry
  - Limits accuracy