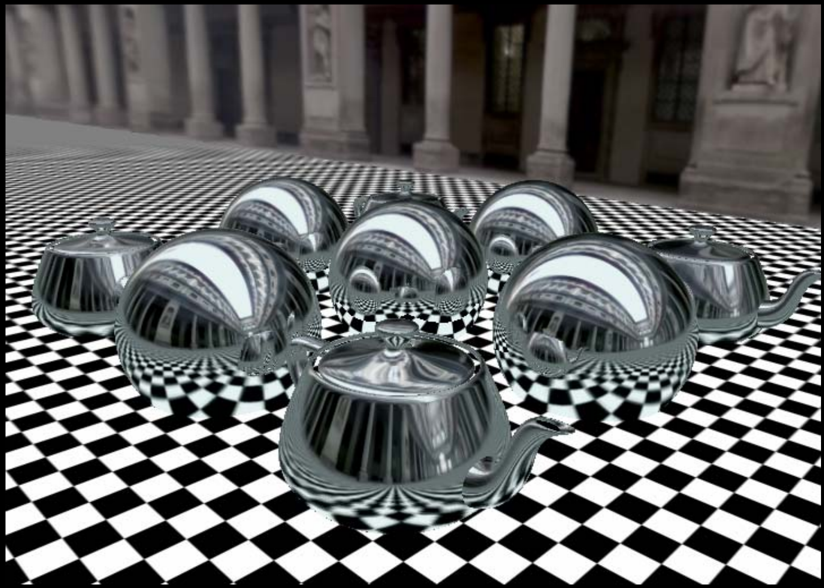
*Voicu Popescu, Chunhui Mei,*
*Jordan Dauble, and Elisha Sacks*
Purdue University

# Reflected-Scene Impostors for Realistic Reflections at Interactive Rates

# Reflections—a difficult problem

- Every reflector is a portal onto a world which is as rich as the directly observed scene and which has complex image formation laws



# Prior work—vast

| | |
|---|---|
| Ray tracing | Image-Based Rendering |
| Feed-forward reflection rendering | Approximation of reflected scene |

# Problem of rendering reflections

- Compute
  - Intersection with reflector
  - Reflected ray
  - Intersection with reflected scene
  - antialiasing

# Problem of rendering reflections

- Compute
  - Intersection with reflector
  - Reflected ray
  - Intersection with reflected scene
  - antialiasing

"OpenGL"

???

# Reflected-scene approximation

- Reflected scene replaced with approx. that provides
  - Fast intersection with ray
  - Antialiasing

# Reflected-scene approximation

- Example: environment mapped reflections
  - Reflected scene infinitely far away
  - Straight forward intersection with ray
  - Antialiasing computed in 2D (mipmapping)

# Reflected-scene approximation

- Example: environment mapped reflections
  - Reflected scene infinitely far away
  - Straight forward intersection with ray
  - Antialiasing computed in 2D (mipmapping)
  - Drastic approximation, incorrect results close to the reflector

# Our approach

- Approximate reflected scene with impostors
  - Considerable prior work on impostors
  - Reflector surface prevents desired viewpoint from getting too close to the impostor
  - Reflection distortion hides impostor artifacts

# Impostor requirements

- Impostor has to provide
  - Fast construction
  - Fast intersection with ray
  - Antialiasing

# Results: billboard impostors

# Results: depth image impostors

# Billboard impostors

- Replace reflected object with billboard
- Higher order reflections
  - Reflective billboards (normal mapped quads)

# Billboard impostors

- Impostor has to provide
  - Fast construction YES
  - Fast intersection with ray YES
  - Antialiasing YES

# Pixel algorithm

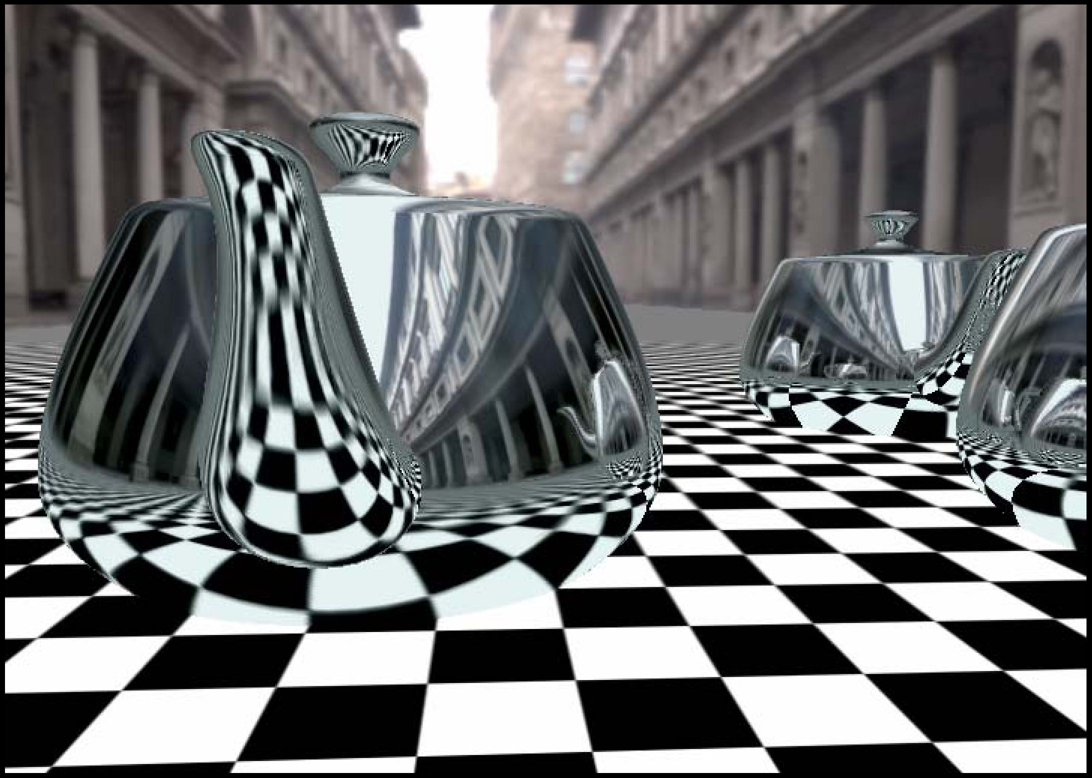
- For *D* diffuse, *R* reflective billboards, and maximum reflection order *K*
  - Compute reflected ray *r*
  - For reflection order 1 to *K*
    - Intersect with (*D*+*R*-1) billboards
    - If no intersection
      - return *EM(r)*
    - Else if intersection with diffuse billboard $DB_i$
      - return $DB_i(r)$
    - Else if intersection with reflective billboard $DB_i$
      - $r = DB_i(r)$

# Pixel algorithm

- For $D$ diffuse, $R$ reflective billboards, and maximum reflection order $K$
  - Compute reflected ray $r$
  - For reflection order 1 to $K$
    - Intersect with ($D$+$R$-1) billboards
    - If no intersect
      - return $EM(r)$
    - Else if interse
      - return $DB_i(r)$
    - Else if intersection with reflective billboard $DB_i$
      - $r = DB_i(r)$

$$O(K*(D+R))$$

# Example: 4 teapots



- $D$ = 1, $R$ = 4, $D+(R-1)+D$ = 5 intersections / pix
- 12 second order reflections
- 40fps

# Example: table scene

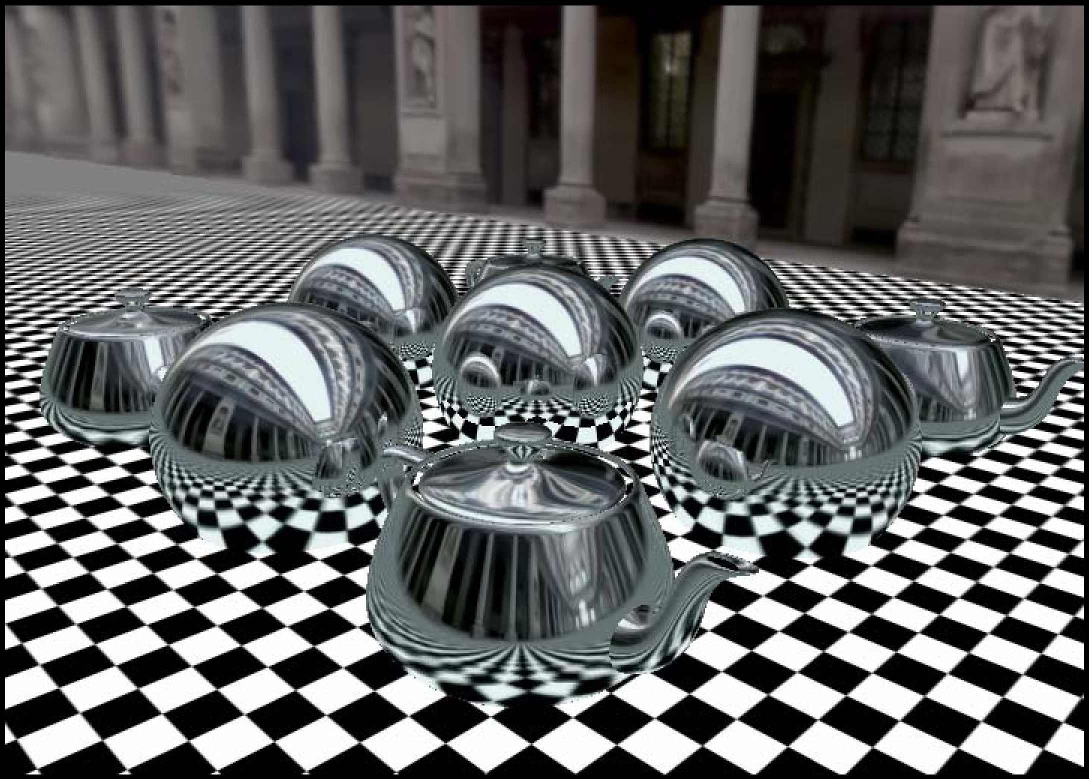
- $D = 2$, $R = 2$, $D+(R-1)+D = 5$ intersections / pix
- 2 second order reflections
- 33 fps

# Example: table scene

- $D = 2$, $R = 2$, $D+(R-1)+D = 5$ intersections / pix
- 2 second order reflections
- 33 fps

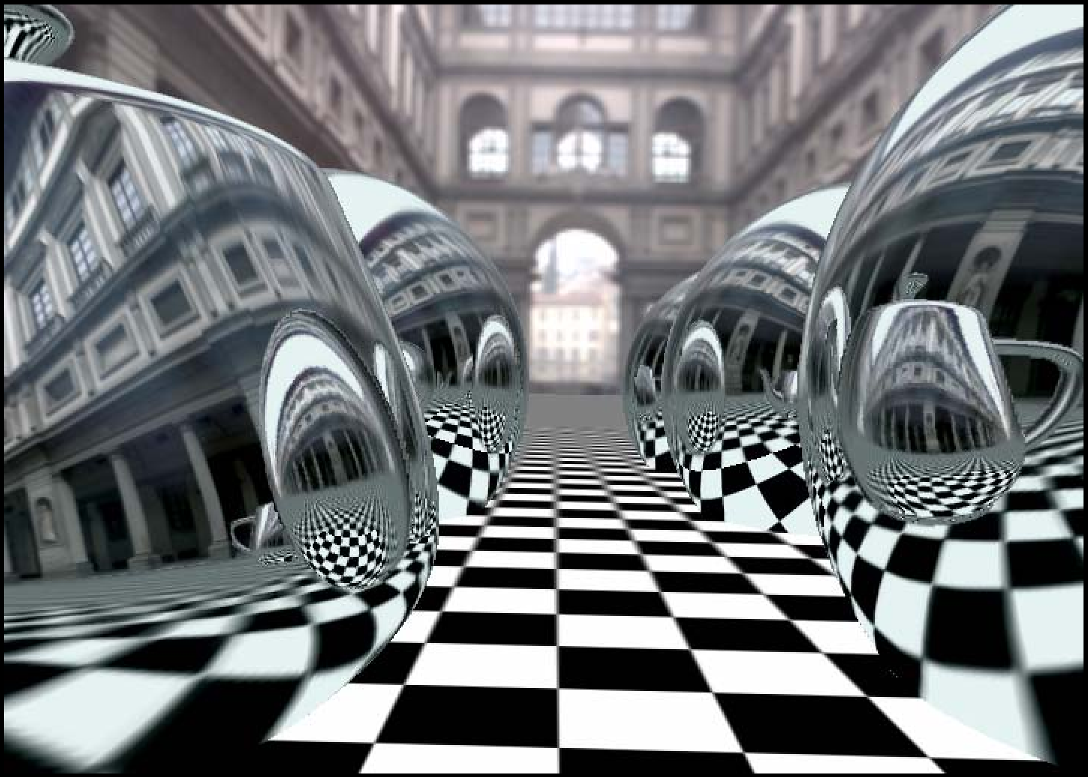
# Example: table scene

# Example: pushing-it scene



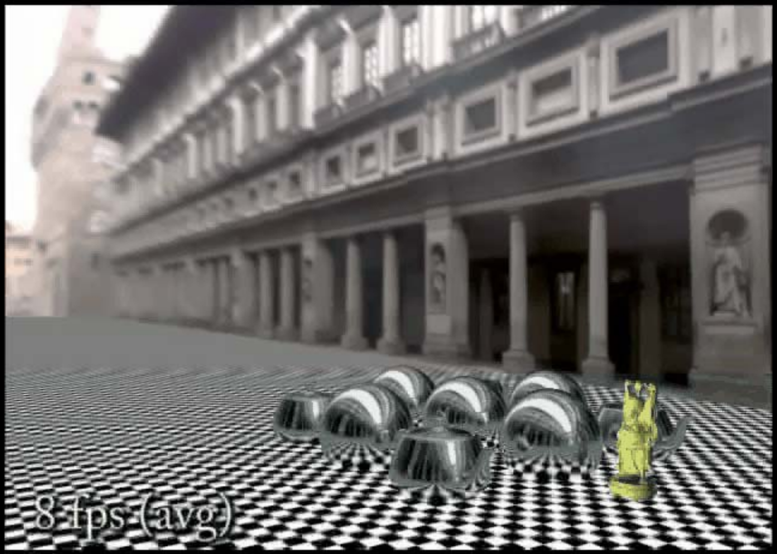
- $D = 2$, $R = 9$, $D+(R-1)+D = 11$ intersections / pix
- 72 second order reflections
- 11 fps

# Example: pushing-it scene

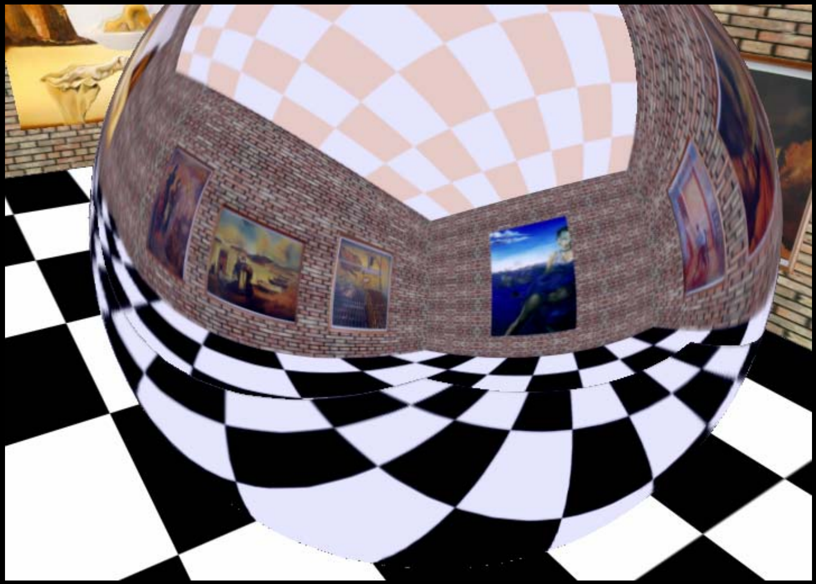
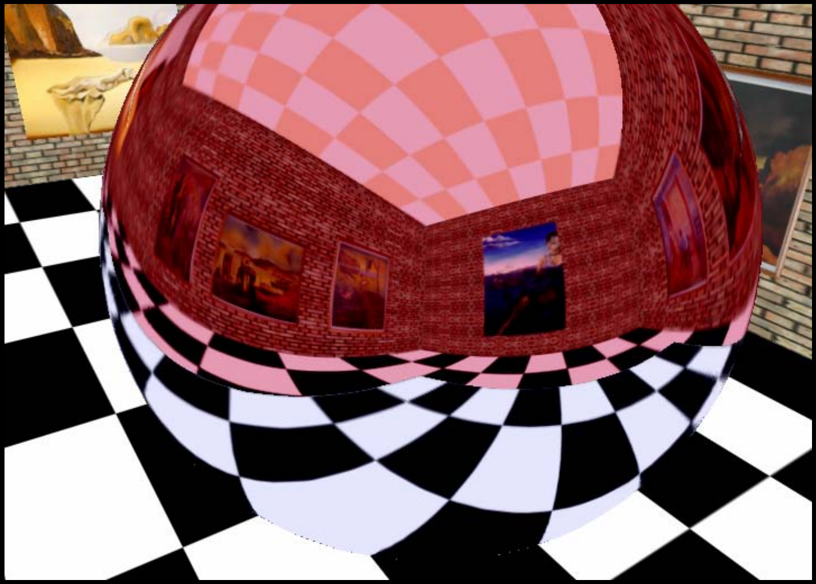


- $D$ = 2, $R$ = 9, $D+(R-1)+D$ = 11 intersections / pix
- 72 second order reflections
- 6 fps

# Example: pushing-it scene

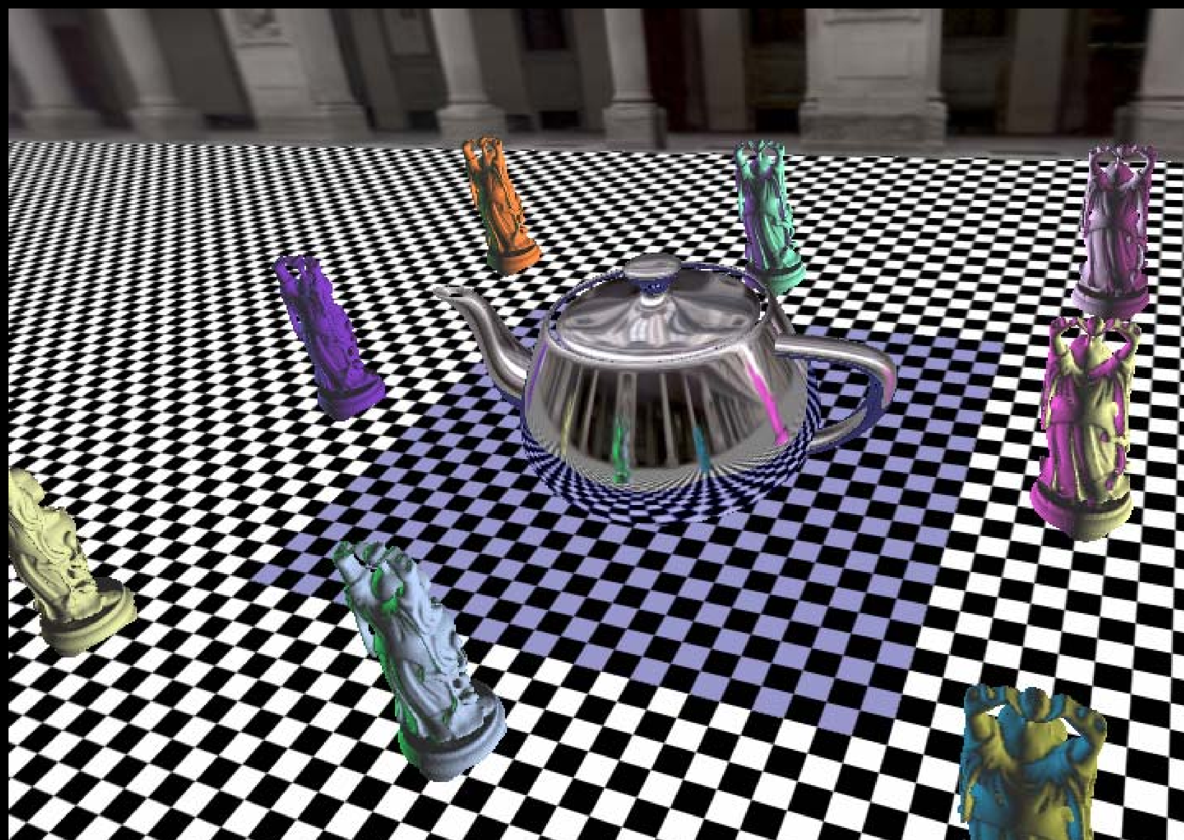


# Problem

Transition from impostor to environment map (*red* in left image) is discontinuous.

# Solution: ray morphing

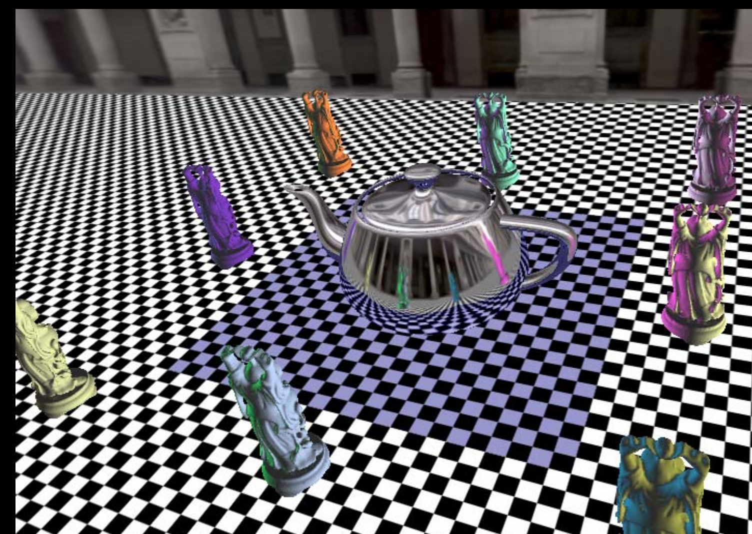




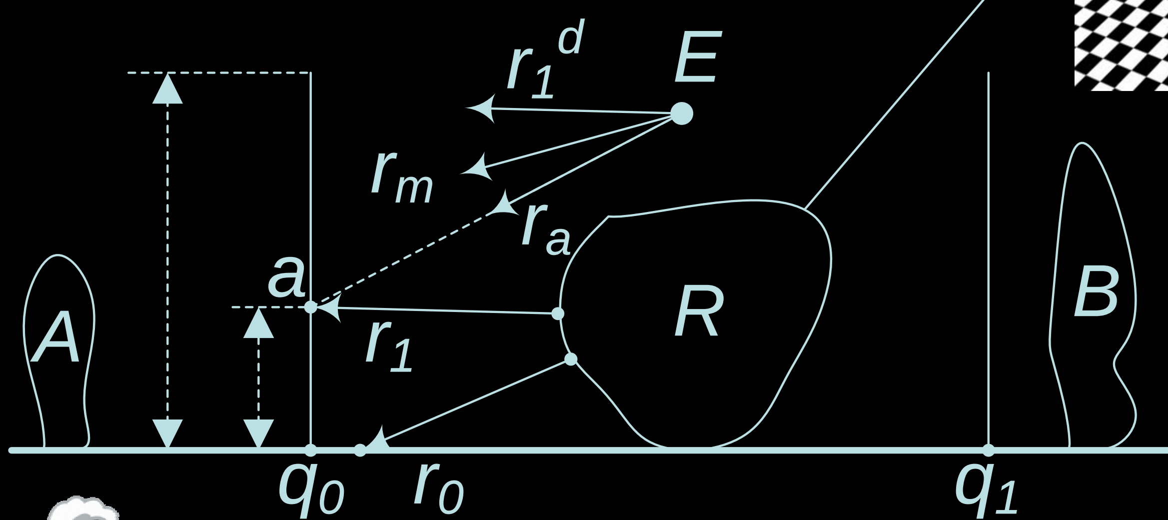
$r_p$ $r_1^d$ $E$ $r_p$ $p$

$r_m$ $j$

# Solution: ray morphing



$$r_m = r_a + (r_1^d - r_a)h/H$$

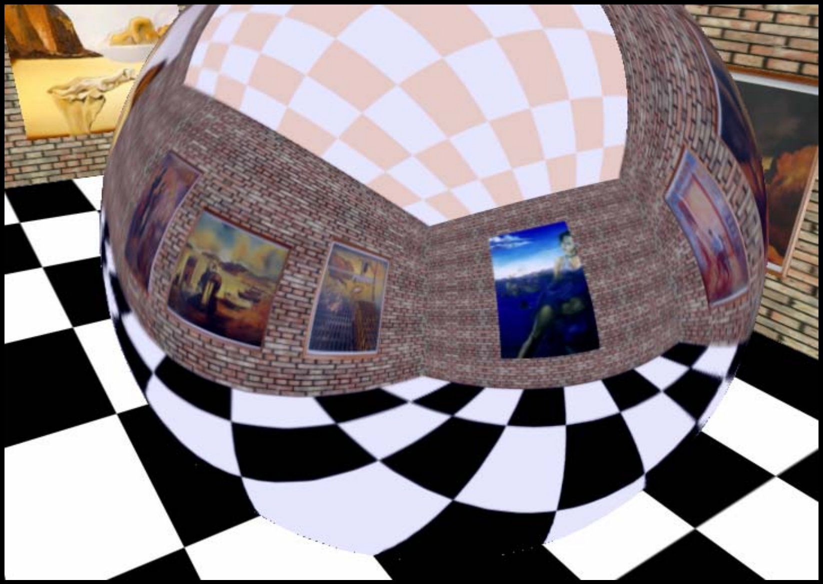





$r_p$ $r_1^d$ E $r_p$ p

$r_m$ j

# Solution




*Left*—continuous transition. *Right*—morph region (*green*), environment map (*red*).

# Ray morphing

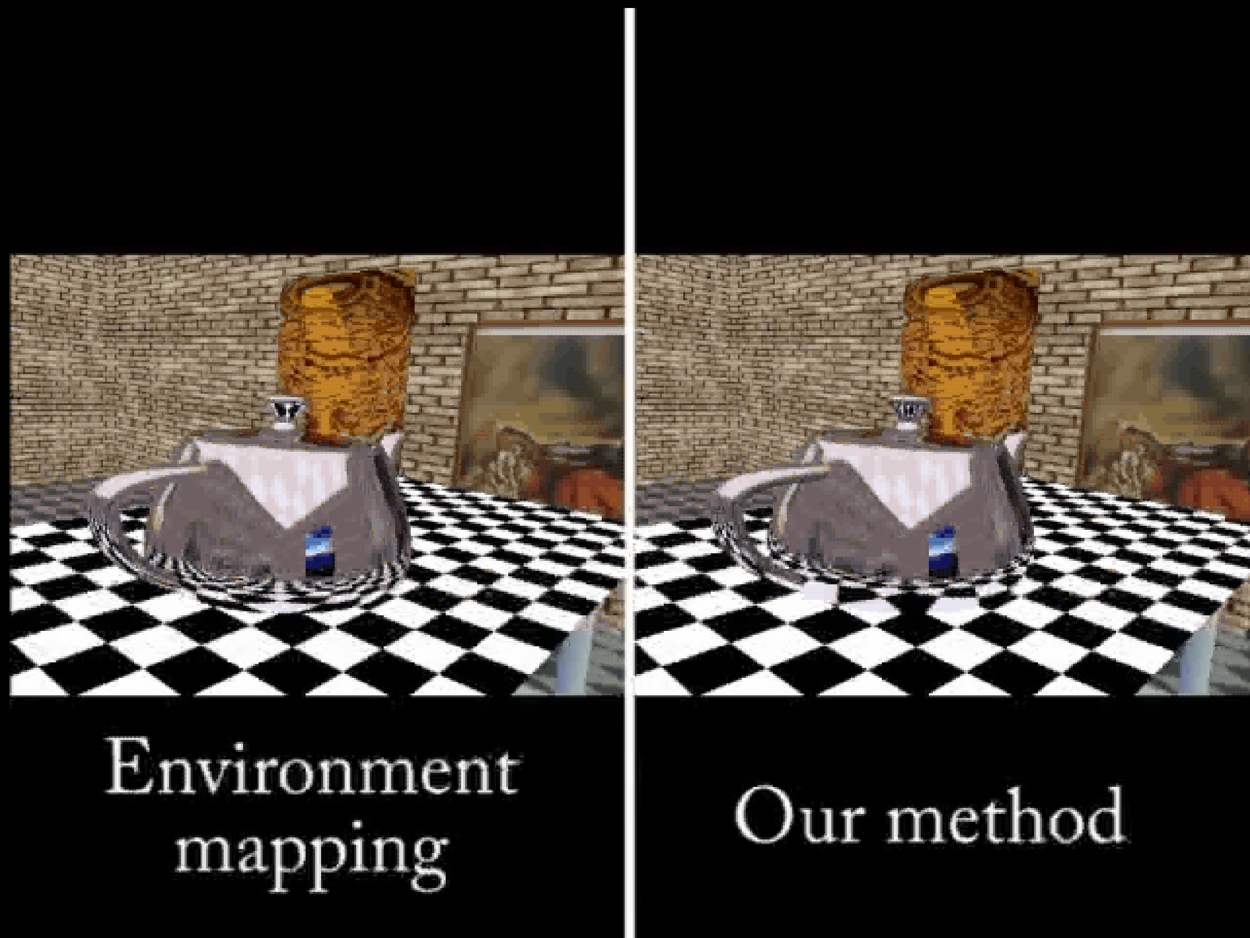
# Attenuation w/ distance

# Fresnel

# Combined effects

# Animation and materials

# Comparison to env. mapping



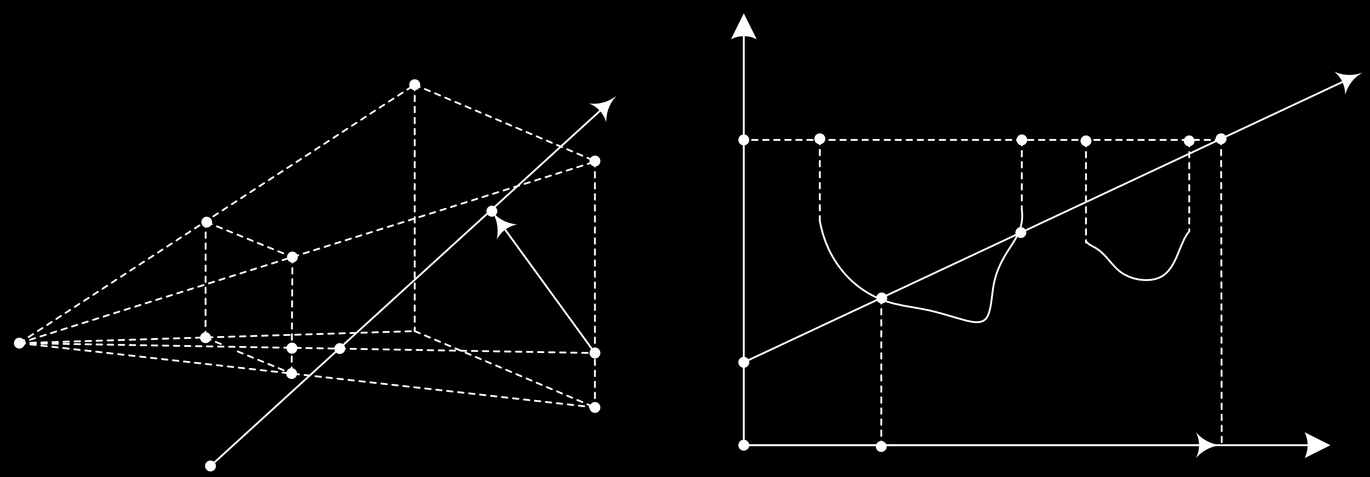
# Billboard limitations

- No support for objects very close to the reflector
- Limited accuracy
  - Flat reflection
  - Lack of motion parallax

# Depth image impostors

- Impostor has to provide
  - Fast construction YES
  - Fast intersection with ray ???
  - Antialiasing YES

# Depth image—ray intersection



Epipolar-like constraints: intersection computed as 1D search
Still too many steps along epipolar segment

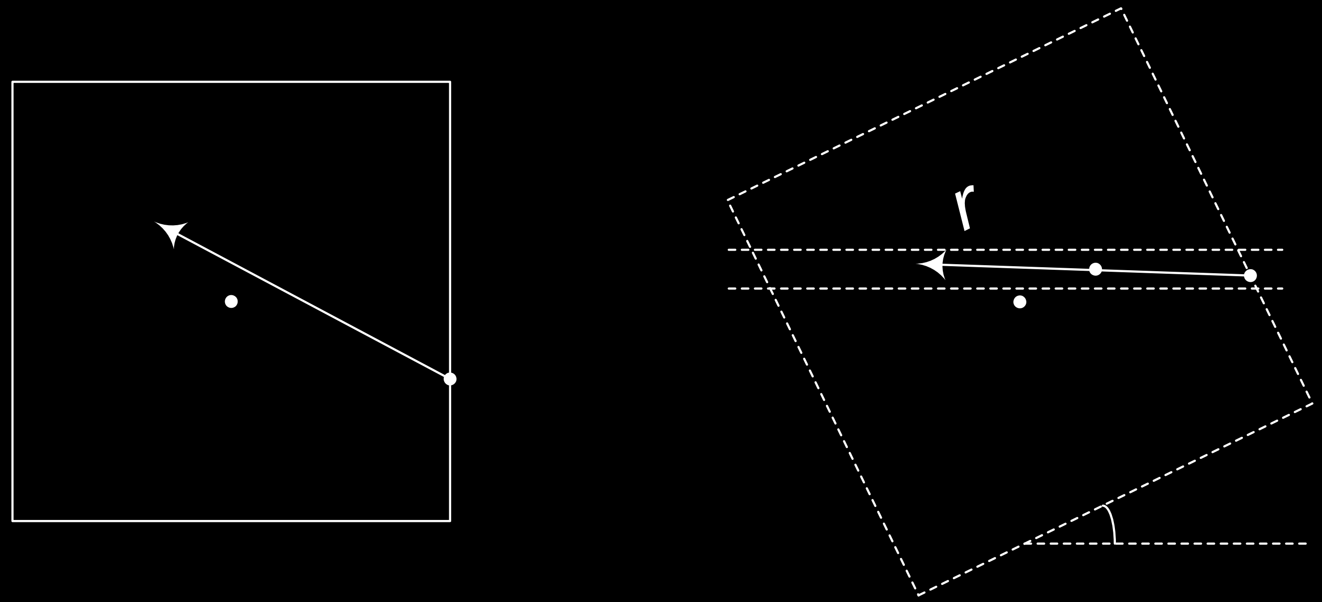
yon

r

hither

b

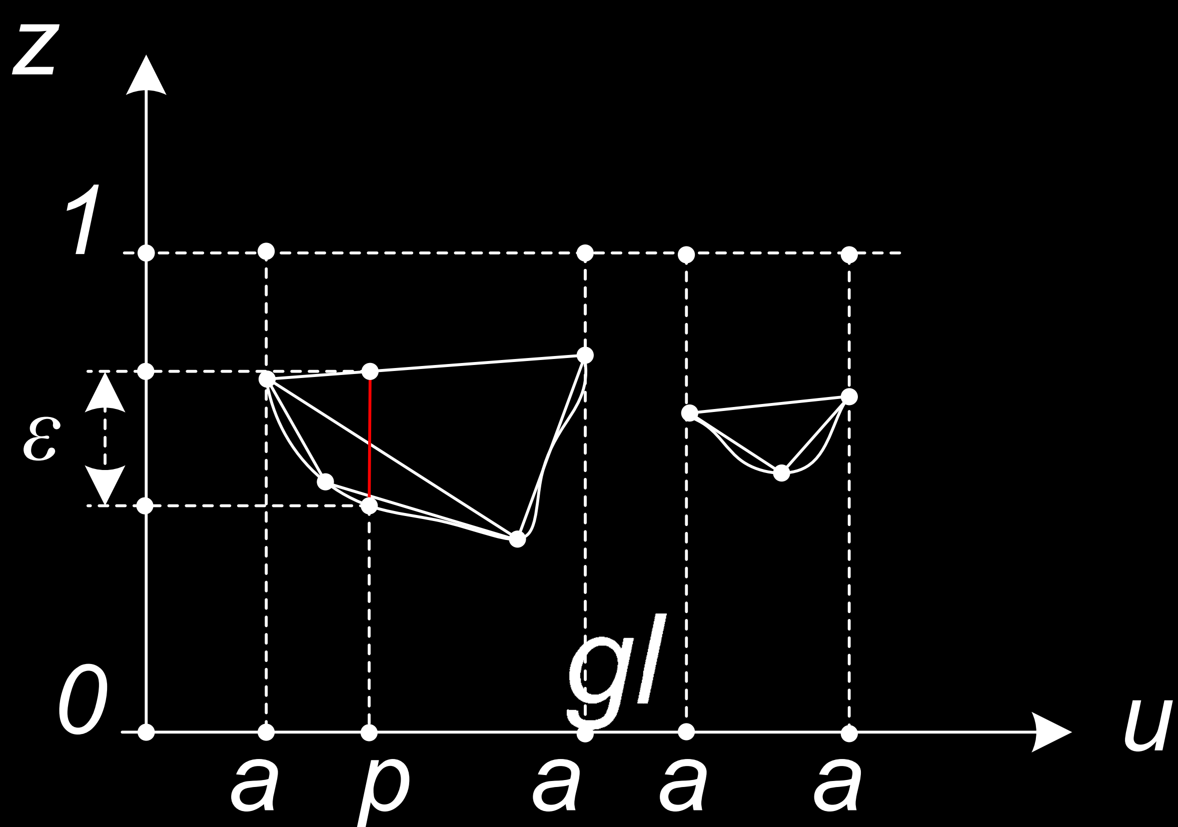
# Simplified Rotated Depth Maps




Pre-rotate depth map.
All rays ever needed project to rows.
Pre-simplify rows.

$r_p$

# Simplified Rotated Depth Maps



# SRDM construction cost

| | | | | |
|---|---|---|---|---|
| *Number of segments* | 8 | 16 | 32 | 64 |
| *Construction time* [ms] | 210 | 300 | 480 | 980 |

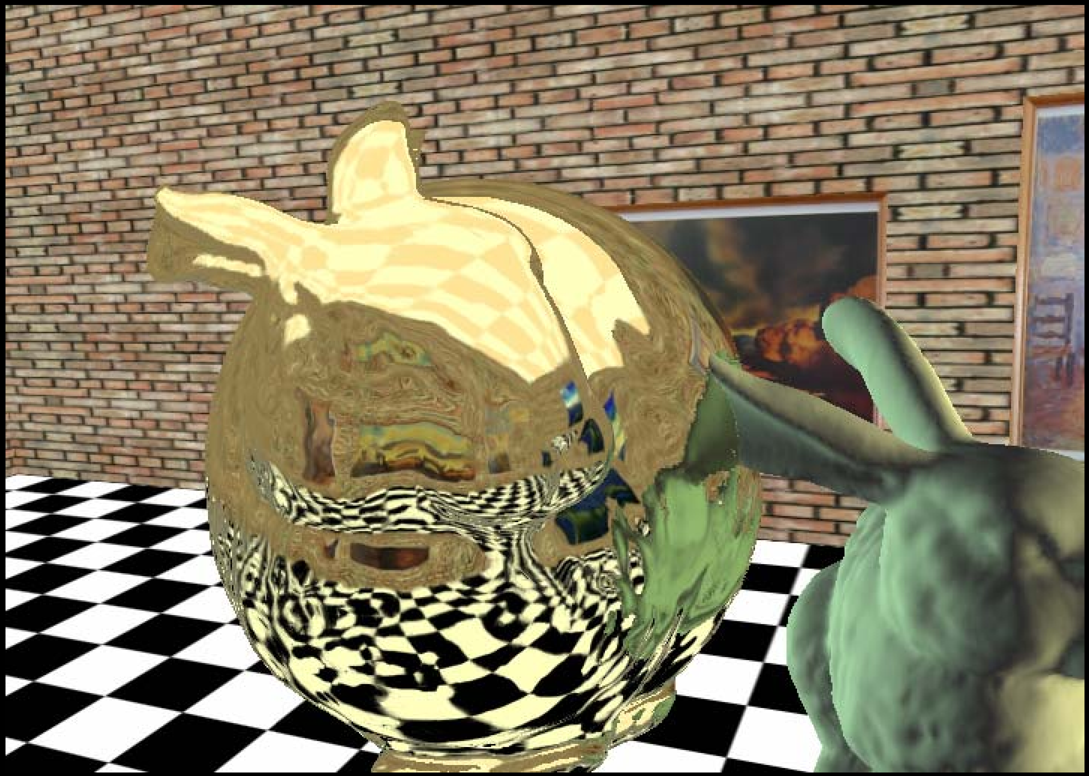
Rigid body transformations, color updates, and reflector updates do not require reconstruction.

# Depth image impostor results

# Depth image impostor results

# Depth image impostor results

# Depth image impostor results



# Depth image impostor results

# Depth image impostor results

# SRDM under-sampling

One rotated depth map every 20$^o$, 10$^o$, 3$^o$, and 2$^o$, respectively.

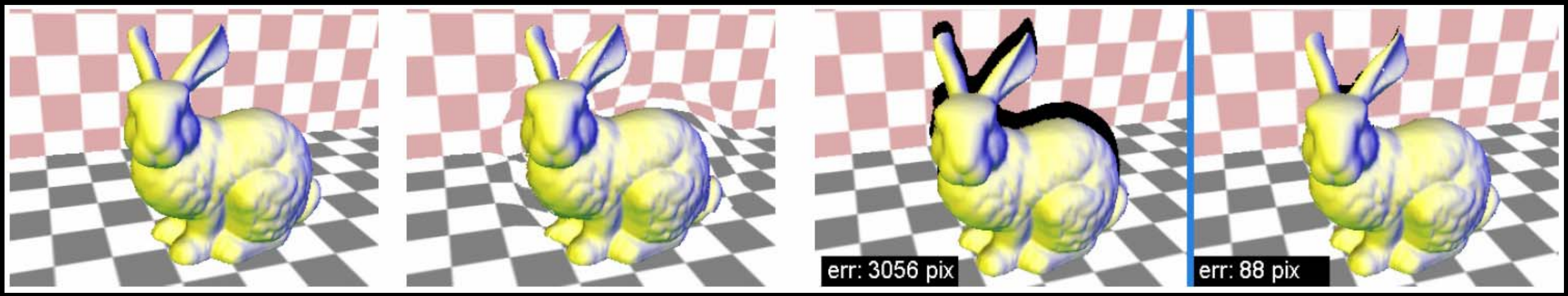
# Depth image impostor results

# Conclusions

- The reflected-impostor approach works
  - Fast, realistic
  - Increased modeling effort
- Rendering reflections reduced to the lesser problem of rendering w/ impostors

# Future work

- Other types of impostors
  - occlusion-resistant



# Future work

- Other types of impostors
- Other BRDFs
- Self-reflections
- Constructing the SRDMs on the GPU

# Acknowledgments


- Funding & equipment
  - NSF, Intel, Microsoft, Computer Science Purdue, Visualization Laboratory Purdue
- Stanford 3D Scanning Rep. for models
- Paul Debevec for environment maps
- Our graphics group at Purdue for miscellaneous but important help