SIGGRAPH2011

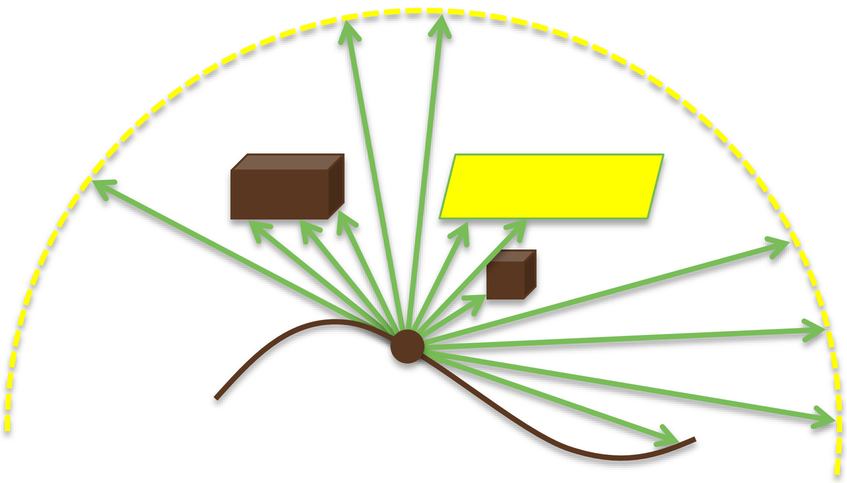
Adaptive Importance Sampling for Multi-Ray Gathering

Ivan Neulander

Rhythm&Hues STUDIOS

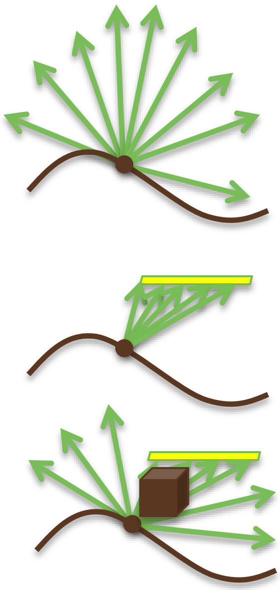
Objective

- Minimize noise during *ray gathering* operation
  - Integrating radiance over solid angle at a fixed position



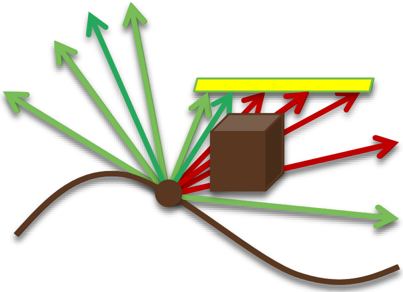
Importance Sampling

- Alter sample choices and weights in order to reduce sample variance
- Commonly based on:
  - BSDF
  - lighting environment
  - Both (MIS)
- Difficult to account for occluding geometry



Adaptive Importance Sampling

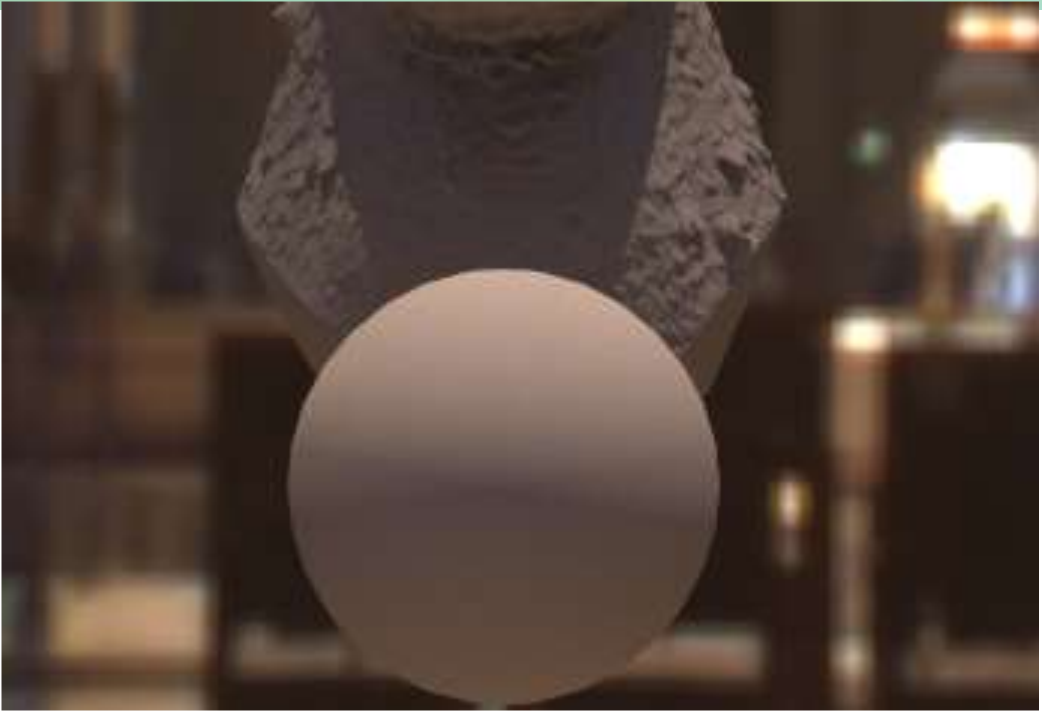
- Allows importance sampler to adapt to ray occlusion
  - Reduces ray density along occluded directions
  - Does not introduce bias
  - Works with other IS schemes



Example: Infinite Area Light

- Uniform
- PHIS
  - Pharr-Humphreys IS
- PHIS + AIS
  - Pharr-Humphreys + Our Adaptive Importance Sampling

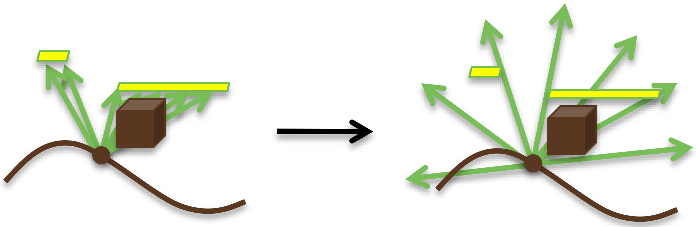
Example: Infinite Area Light



PhilS rmAIS

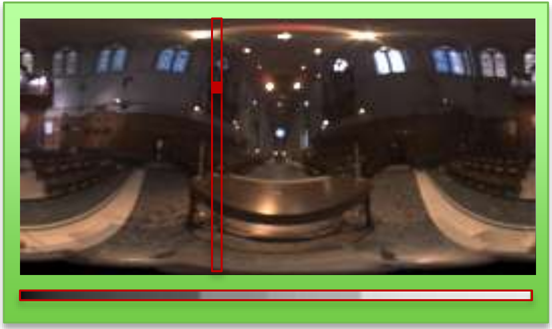
Initial Idea #1

- For each occluded ray, adjust MIS to favor BSDF over lights
  - Does not generally reduce noise
  - Usually increases it



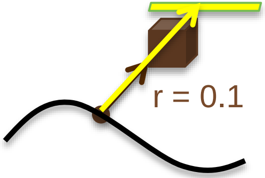
Initial Idea #2

- Modify cdfs of PH sampler dynamically based on ray-sampled directions
  - Does reduce noise, but:
  - Too expensive for large maps
  - Does not generalize to arbitrary area lights

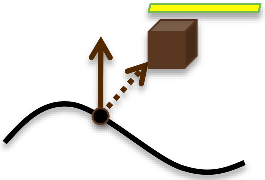


Our Approach

- Each ray is rated by the renderer
  - Compares actual radiance to unoccluded radiance
- Rating is incorporated into an *affinity map*
  - Spherical mapping: direction → pixel coordinate
- Future rays are stochastically accepted or rejected based on the affinity map



- New batch of rays starts with empty affinity map



Affinity Map

- Multi-Resolution spherical texture
  - Larger pixels → less variance but less directional specificity
  - We use highest resolution where affinity < 1 (else 1)
- 3 float channels:
  - [affinity sum, weight sum, reset counter]
  - [affinity sum, weight sum] start at [1, 1] → biased toward high affinity

0 rays | 32 rays | 128 rays

Querying and Updating Affinity Map

Minimum affinity — Mean affinity

$$\text{affinity}(\theta,\phi) = a_{min} + \frac{\text{map}(\theta,\phi)[0]}{\text{map}(\theta,\phi)[1]}(1 - a_{min})$$

$$a_{min} = 1/(tol \cdot \underbrace{height^2_{map}}_{\text{"penalizes" large pixels}})$$

weight

$$w = 1/\text{affinity}(\theta,\phi)$$

$$\text{map}(\theta,\phi)[0] \mathrel{+}= rw$$

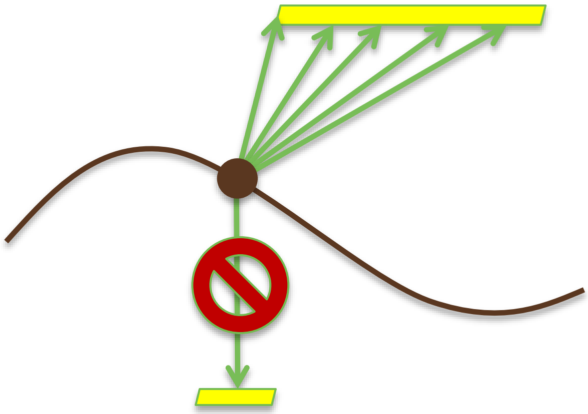
$$\text{map}(\theta,\phi)[1] \mathrel{+}= w$$

Stochastic Ray Rejection

- Russian Roulette:
  Each ray is stochastically accepted or rejected based on its *affinity* value
- Rejected ray is skipped without tracing or shading
  - Does not count toward desired ray total
- Accepted ray is traced and shaded
  - Counts toward desired ray total
  - Weight is scaled by 1/*affinity*

Unconditional Ray Rejection

- Rays outside BSDF support are always rejected
  - Do not count toward desired ray total
  - No weight adjustment following non-rejection



Adaptive Sampling

- We continue drawing rays from a batch until *n* are accepted
- We count rejected rays as zeros but do not sample them
- Two problems:

1. Sample stratification is tricky
   - Unstratified sampling is noisy
2. Selection bias
   - Average for ray batch is biased toward high-affinity directions

Sample Stratification

- Two issues:
  1. Total sample count is unknown
  2. Only a random subset of samples is used
- Simple random sampling works, but is noisy
- Our approach: use 3-D Halton sequence for sample placement and rejection

simple

Halton

Selection Bias: Problem

- Analogy:
  Country where people continue having children until *n* boys (50% chance of boy vs. girl)
  - Child ➔ ray
  - Boy ➔ accepted ray
  - Family ➔ batch of ray

e.g. $n = 2$ BGB GBGB BGGGB …

- Average among all children : 50% boys (unbiased)
- Average family: >50% boys (biased) BB GGGGGGGGBB
- Need family average to be unbiased
  - All families carry equal weight in our census

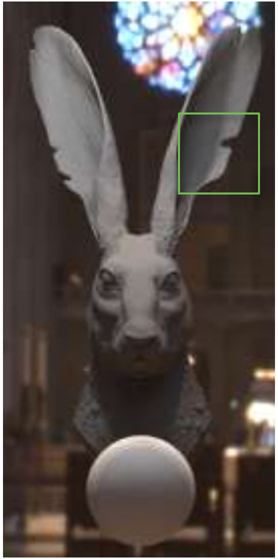
Selection Bias: Solution

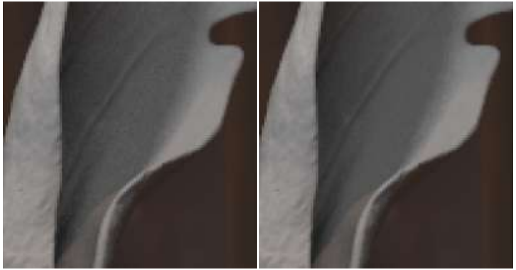
- To remove bias:
  Continue having children until n+1 boys but reject last boy
- Back to gathering a batch of n accepted rays:
  - Keep sampling until n+1 accepted rays
  - Count all rejected rays after n rays
  - Ignore the last (accepted) ray
- E.g. $n = 3$, $p = 25\%$
  - Biased: 0010000101
  - Unbiased: 0010000101001

ignored

Shadow Edge Problem

- Russian roulette can produce noise



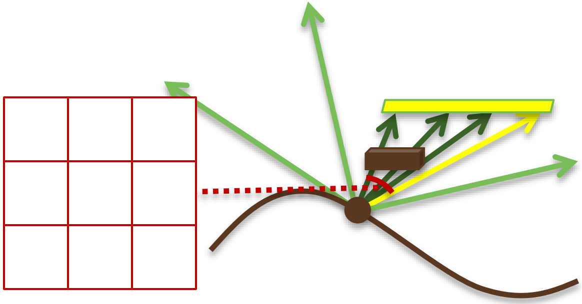


off aggressive

AIS setting

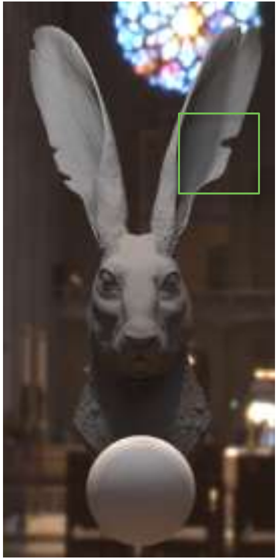
Shadow Edge Solution

- Conservative Rejection:
  - any increase in affinity → force high affinity at and around current pixel (1-pixel border)
  - Wastes more rays on “dark” directions…
  - …but avoids missing rays on “bright” directions
- Example (tol = ∞ ➔ a_{min} = 0)
  - Affinity =
  - 1 / 1 = 1.0
  - 1 / 2 = 0.5
  - 1 / 4 = 0.25
  - 1 / 8 = 0.125
  - 1 (forced)



Shadow Edge Solution: Result

- Good balance of noise reduction at shadow interior vs. edges



off aggressive conservative

AIS setting

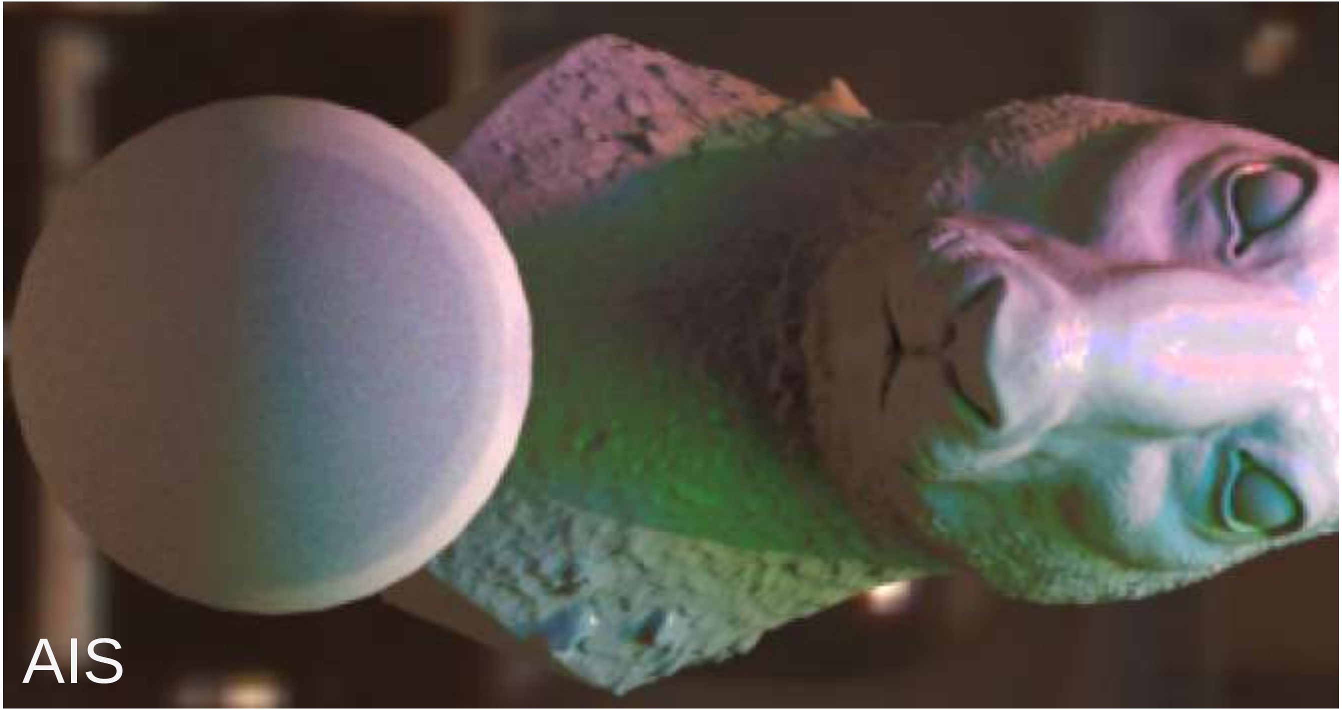
Results: render times match

- Model: 400k tri
- Lighting:
  - Infinite area light with HDRI texture
  - 2 sphere area lights
  - 1 plane area light

	Reference	AIS
Nominal rays/pixel	158	116
CPU time per frame	174.9s	174.8s

Results: render times match

SIGGRAPH2011

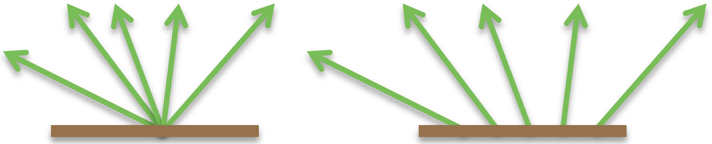


Performance Considerations

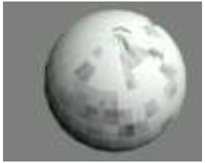
- Affinity map overhead
  - Speed: 2-5%
  - Memory: < 100KB typically
- Time cost of ray rejection: depends
  - Benefits from efficient underlying IS
  - No rejection sampling → expensive to adjust IS profile on the fly
- Additional rays being traced and shaded: depends
  - Accounts for most of the render time increase with AIS
  - But these rays tend to contribute significantly to surface irradiance

Future Work

- Allow for perturbation of ray origins in batch



  - Idea: bias affinity toward 1 as origins diverge
- Automatically disable AIS in some cases
  - E.g. giant penumbra
- Improve parametrization and filtering of affinity map
  - Point-sampled lat-long map is fast, but not ideal
  - Use less distorting mapping, bilinear filtering



Conclusion

- Thank you to Rhythm & Hues
  - Keith Goldfarb
  - Kevin Beason
  - Chris Rogers
  - Ryan Gillis
- This talk: www.neulander.org/work#sketch2011