

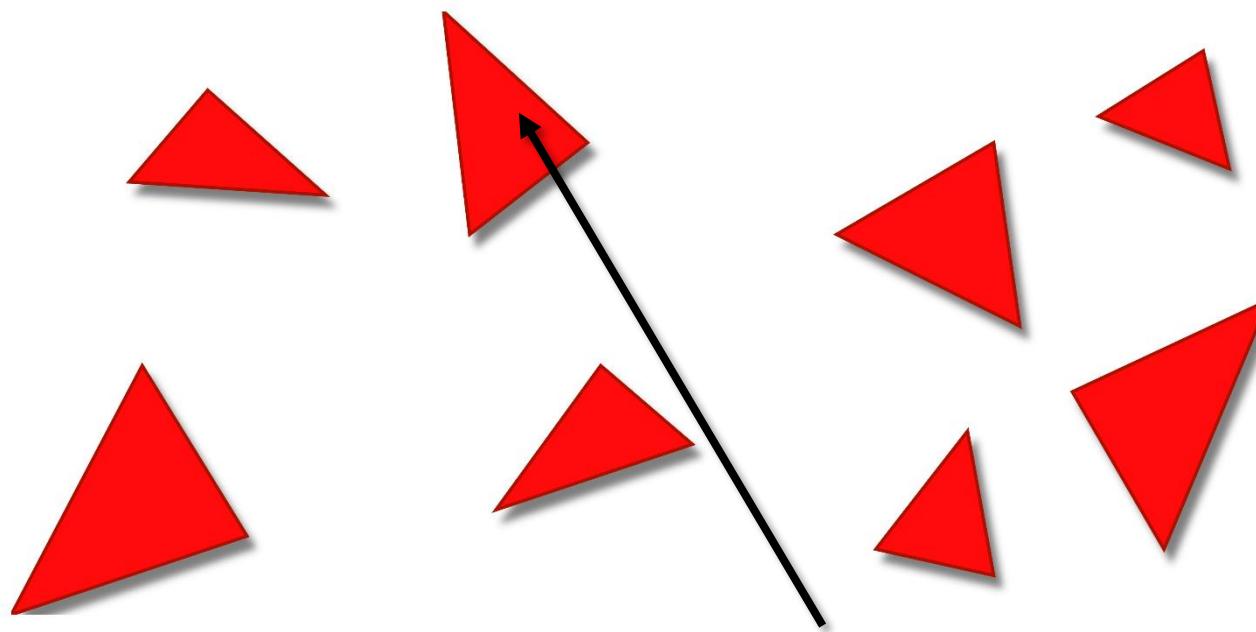


Efficient visibility heuristics for kd-trees using the RTSAH

Matthias Moulin, Niels Billen and Philip Dutré

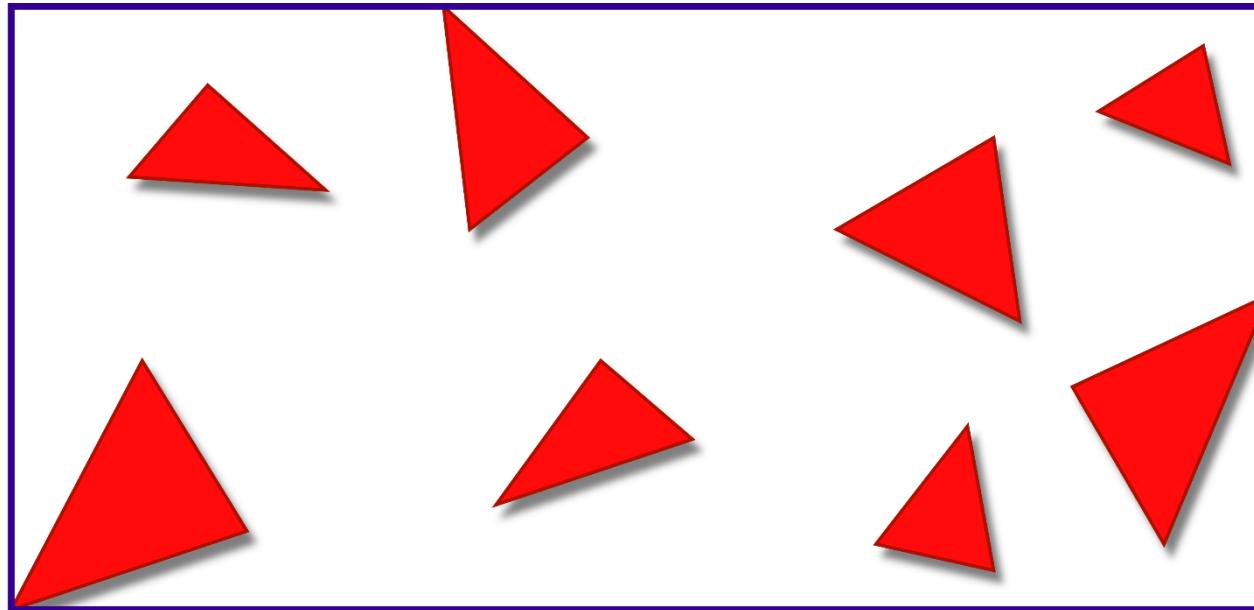
Department of Computer Science
KU Leuven, Belgium

kd-trees



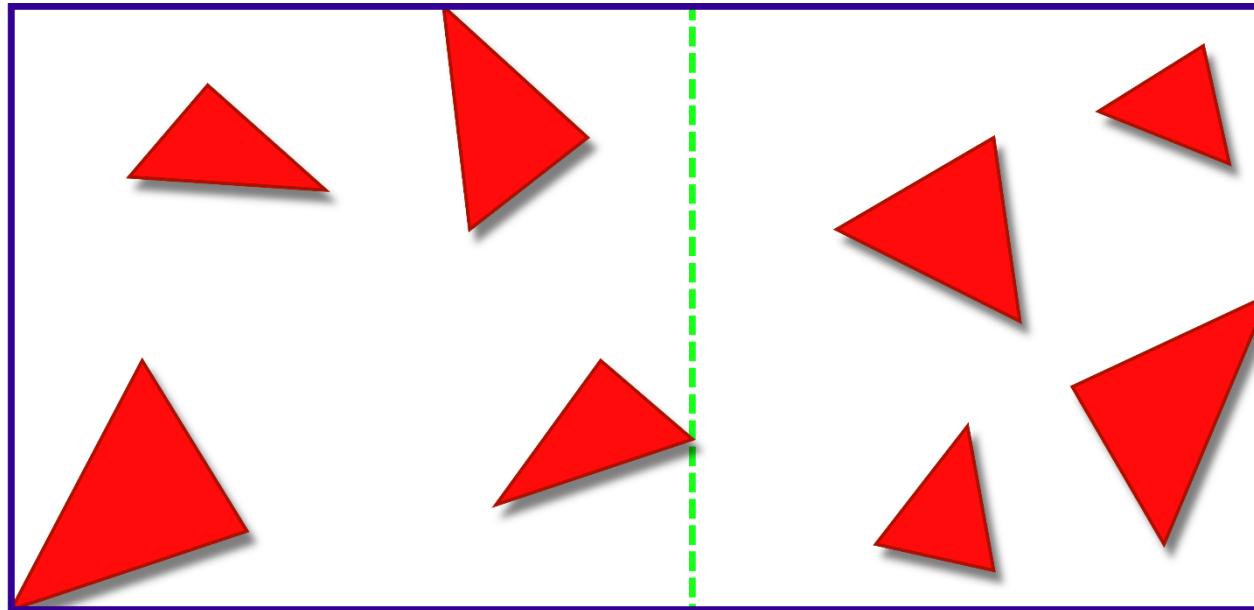
kd-trees

Recursively split 3D space



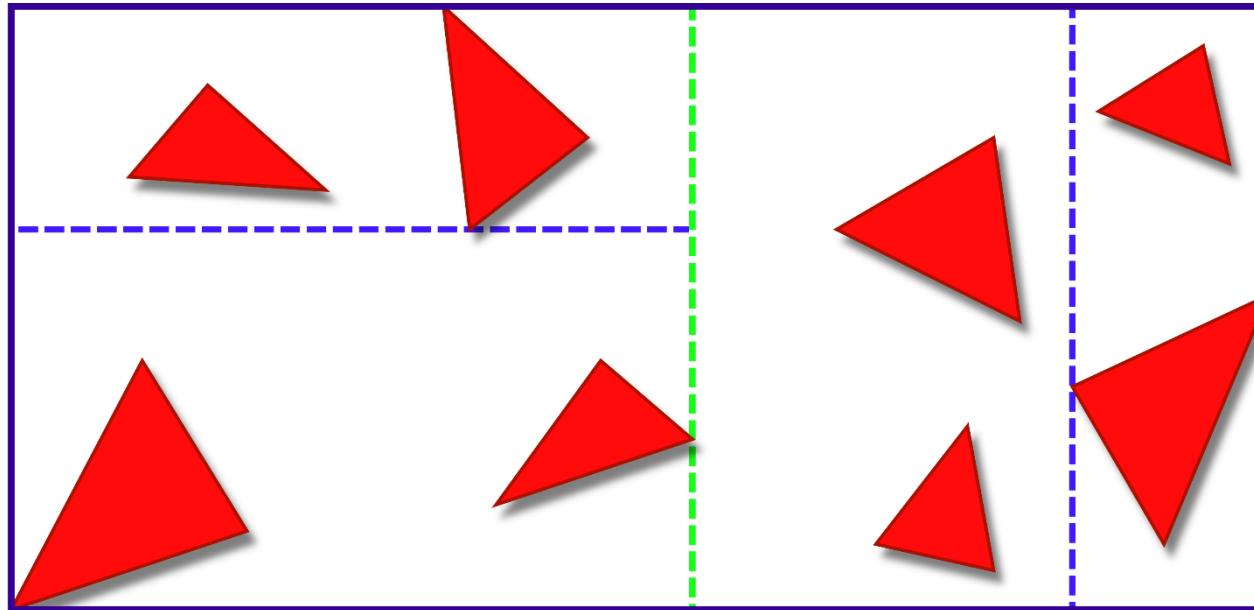
kd-trees

Recursively split 3D space



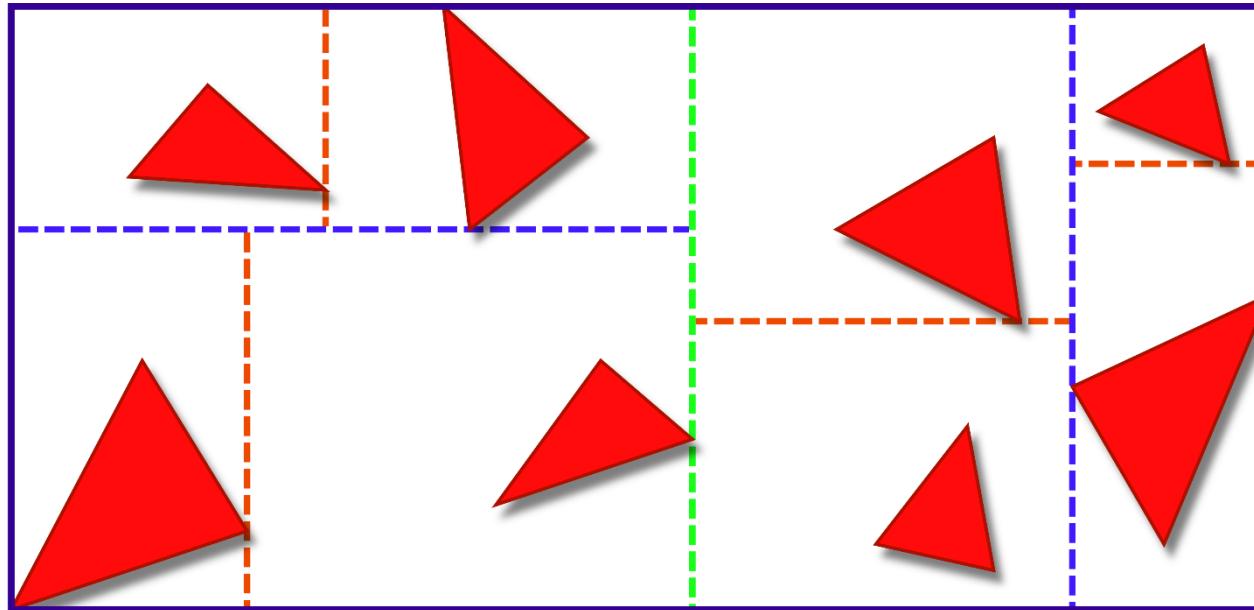
kd-trees

Recursively split 3D space



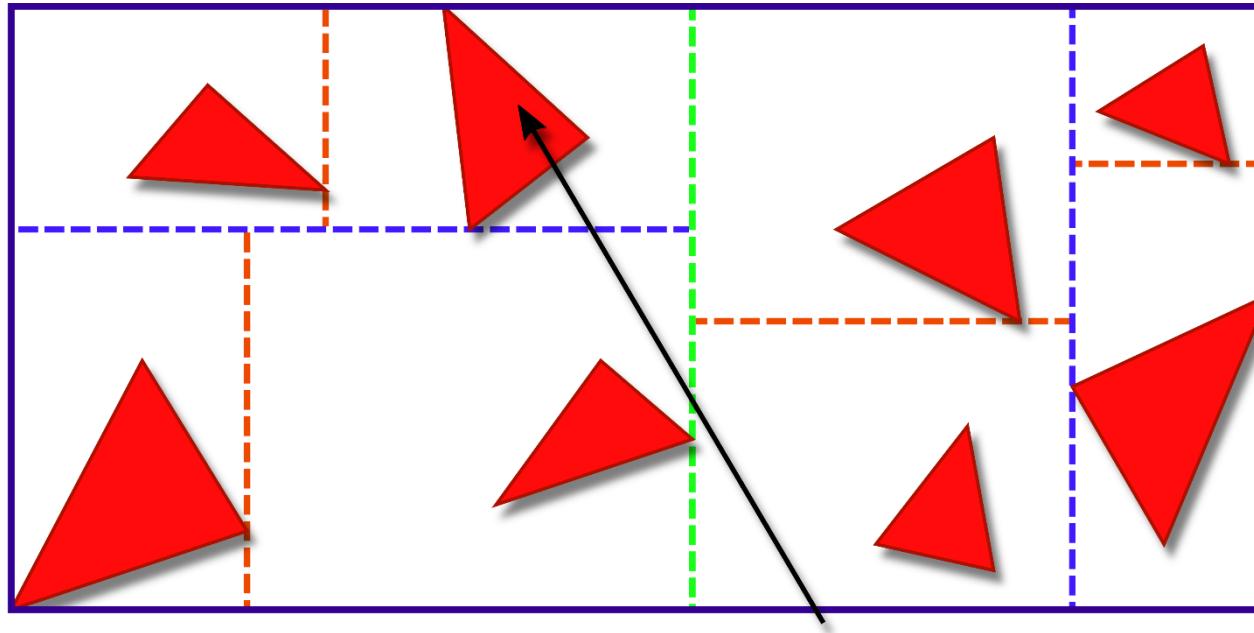
kd-trees

Recursively split 3D space



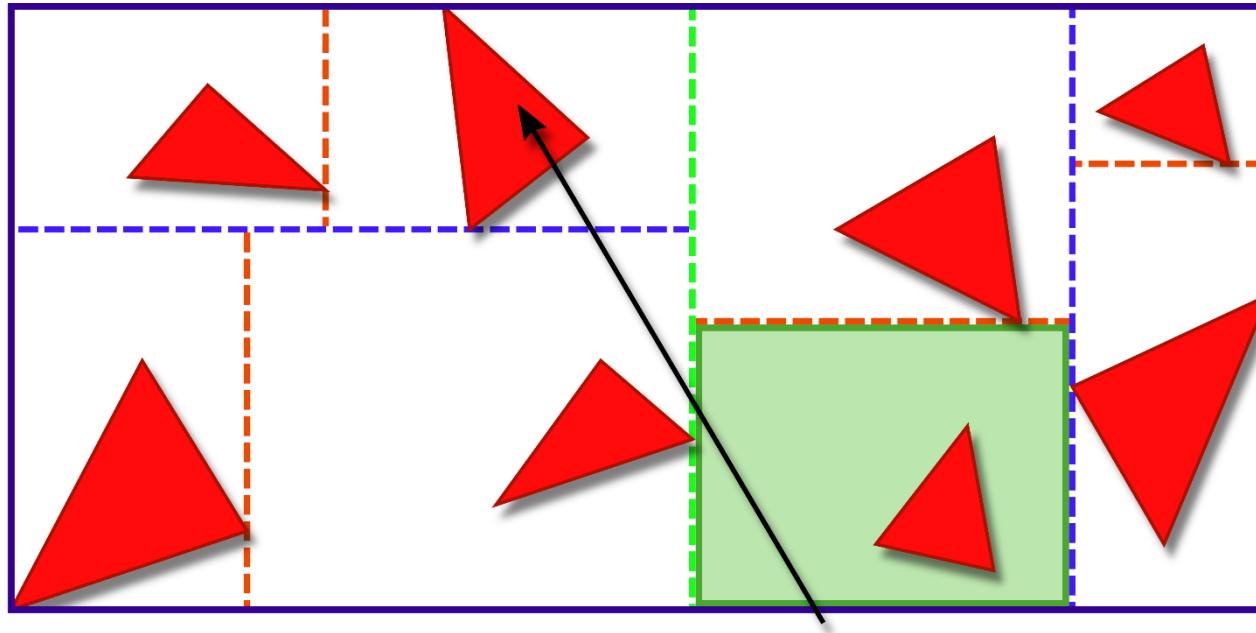
kd-trees

Trace rays front to back



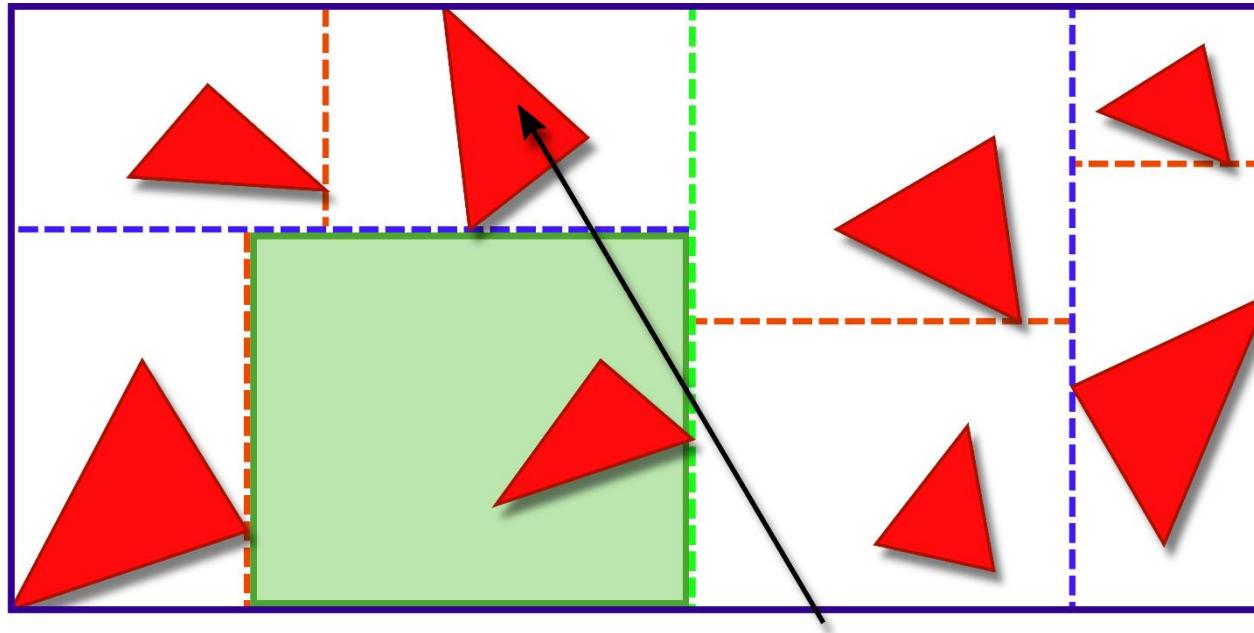
kd-trees

Trace rays front to back



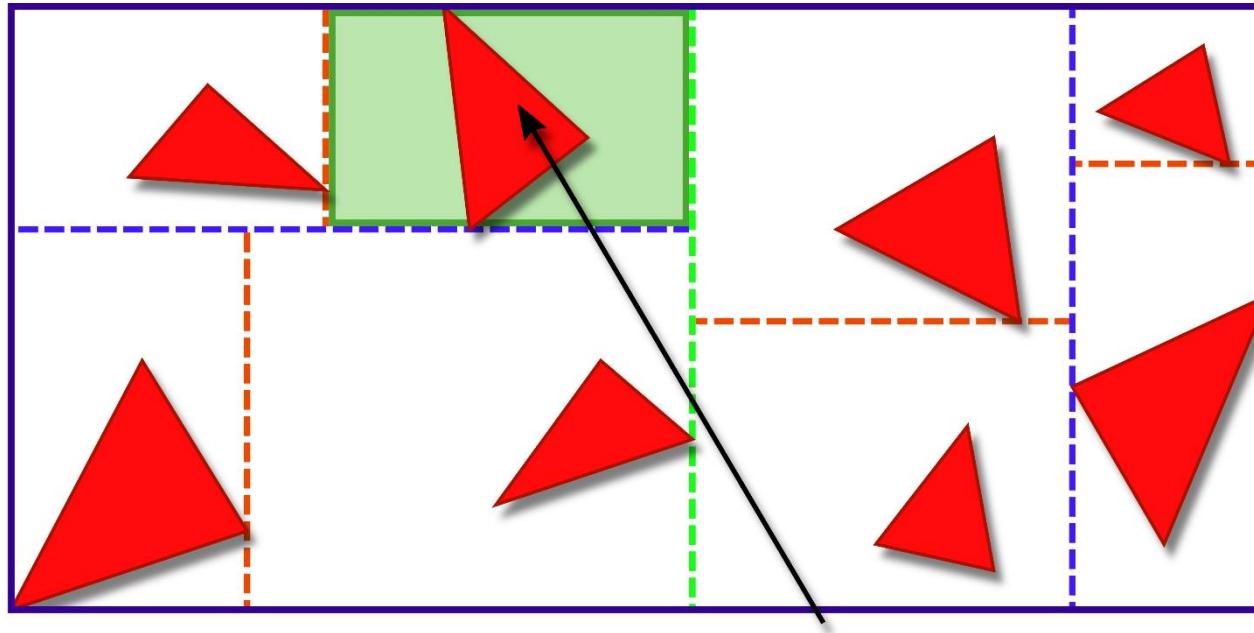
kd-trees

Trace rays front to back



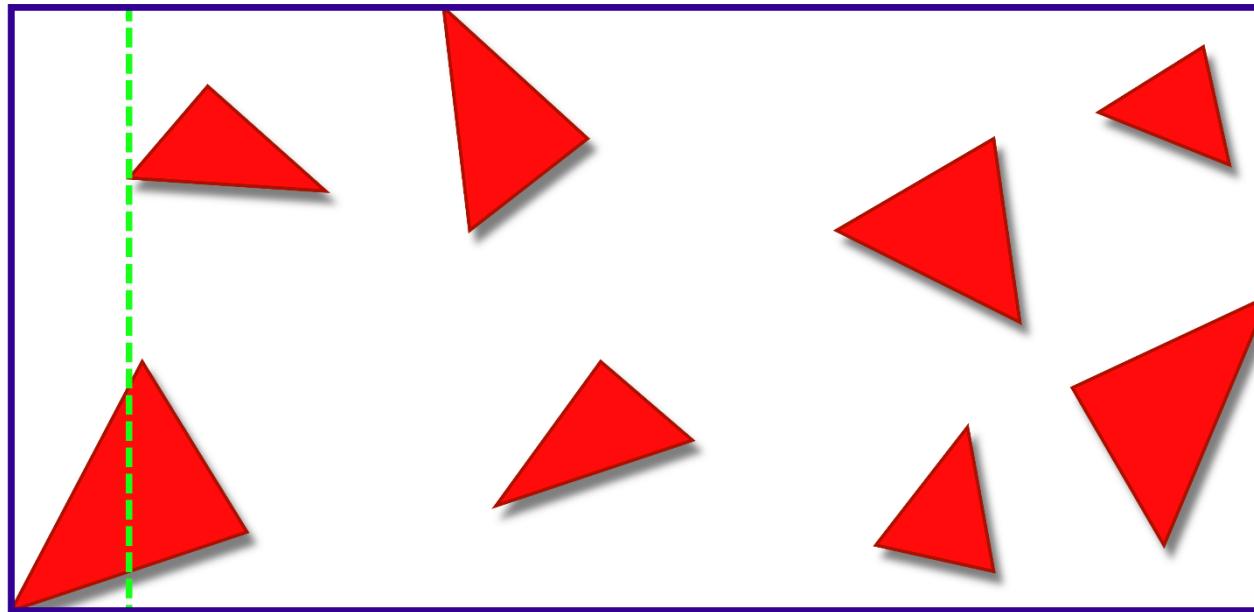
kd-trees

Trace rays front to back



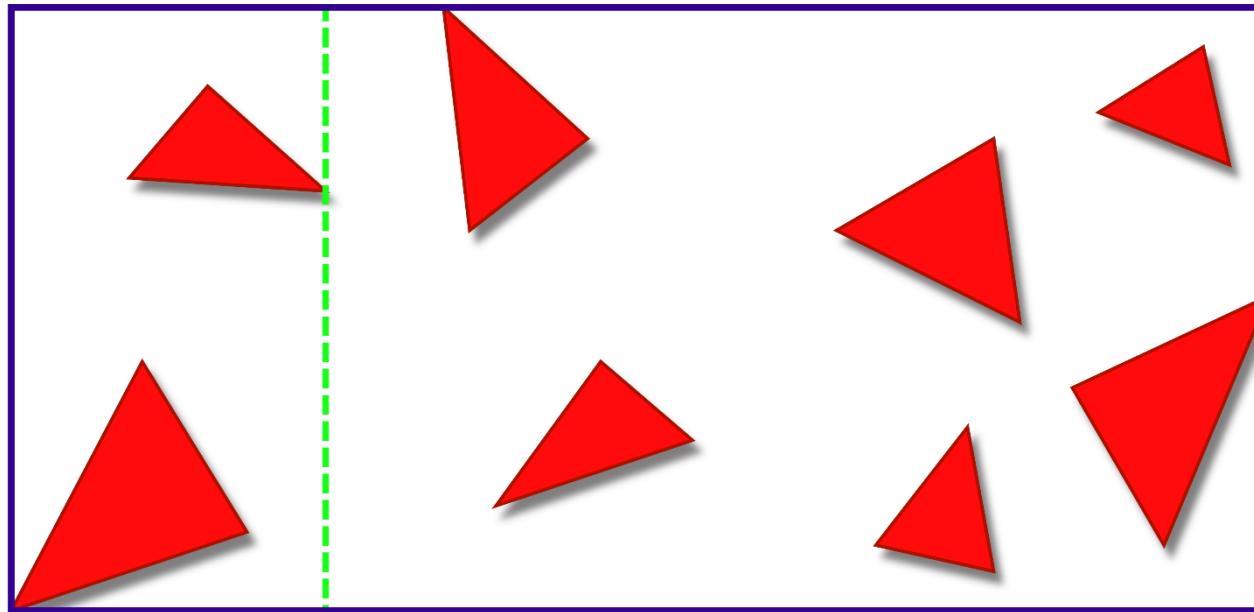
kd-trees

Problem: where to put the splitting planes?



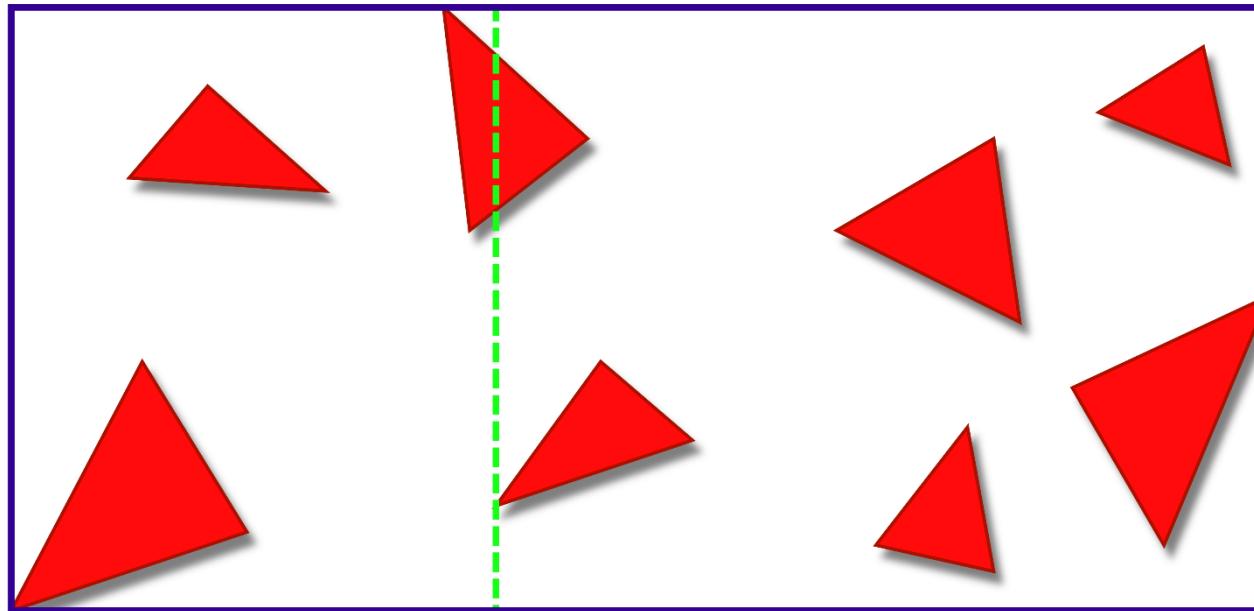
kd-trees

Problem: where to put the splitting planes?



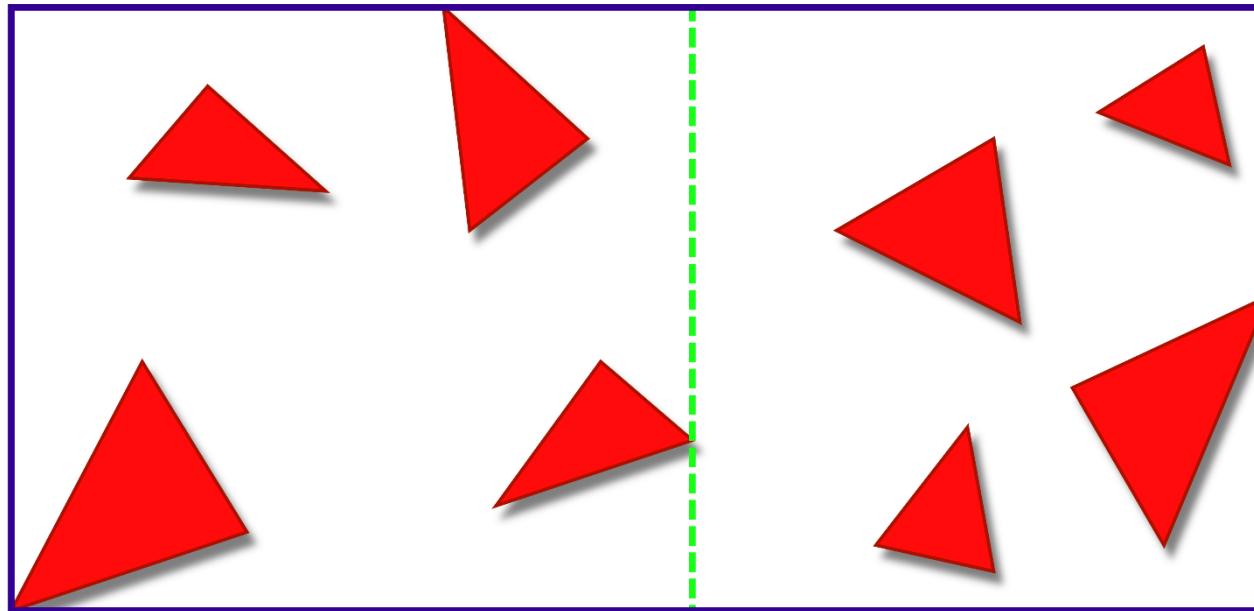
kd-trees

Problem: where to put the splitting planes?



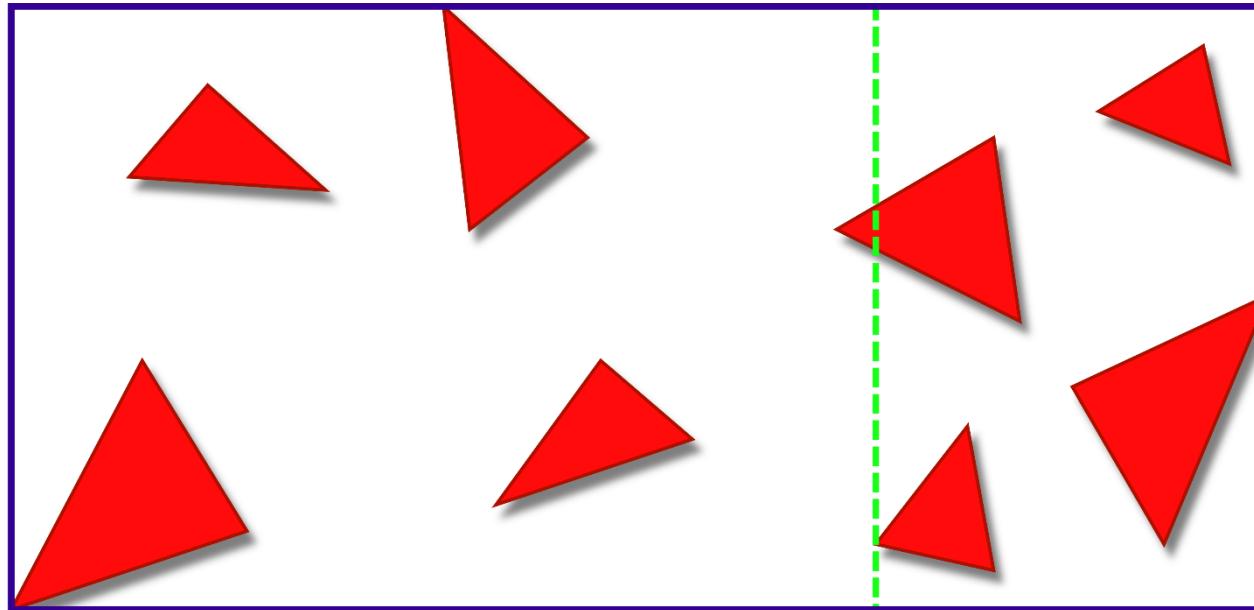
kd-trees

Problem: where to put the splitting planes?



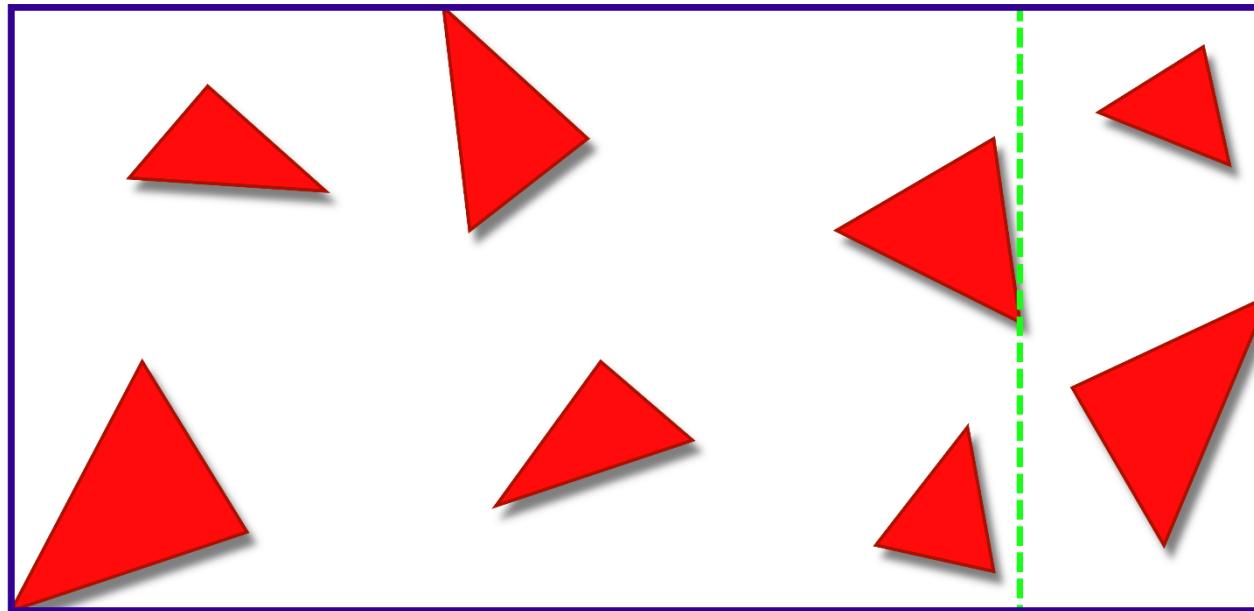
kd-trees

Problem: where to put the splitting planes?



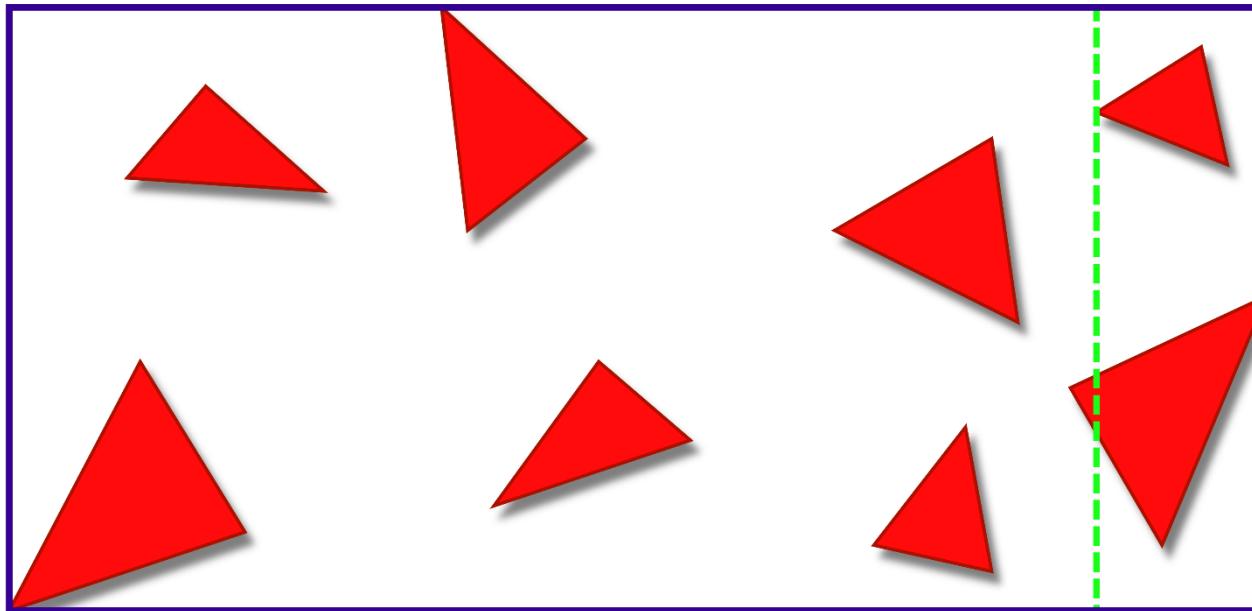
kd-trees

Problem: where to put the splitting planes?



kd-trees

Problem: where to put the splitting planes?



Literature

- **Surface Area Heuristic**

[Goldsmith & Salmon 1987], [MacDonald & Booth 1990]

- **Optimizations**

- $\mathcal{O}(N \log(N))$ build procedure
- Mail-boxing

[Wald & Havran 2006]

[Hunt 2008]

- **Alternative build heuristics**

- Interior rays
- Actual ray distribution
- Non-uniform ray distribution
- Blocking factors

[Fabianowski et al. 2009]

[Havran & Bittner 1999], [Bittner & Havran 2009]

[Choi et al. 2012]

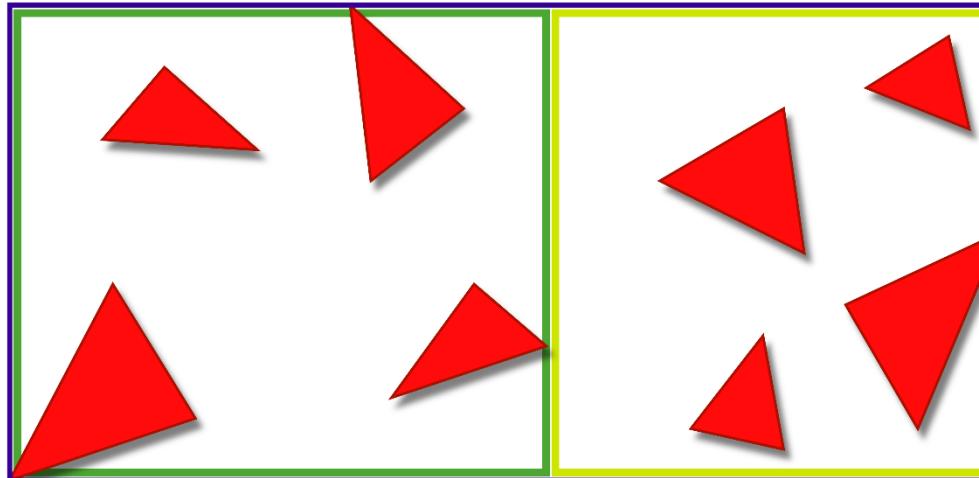
[Reinhard et al. 1996], [Havran 2000]

Surface Area Heuristic

Where to put the splitting plane?

→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + p_L C_L + p_R C_R$$

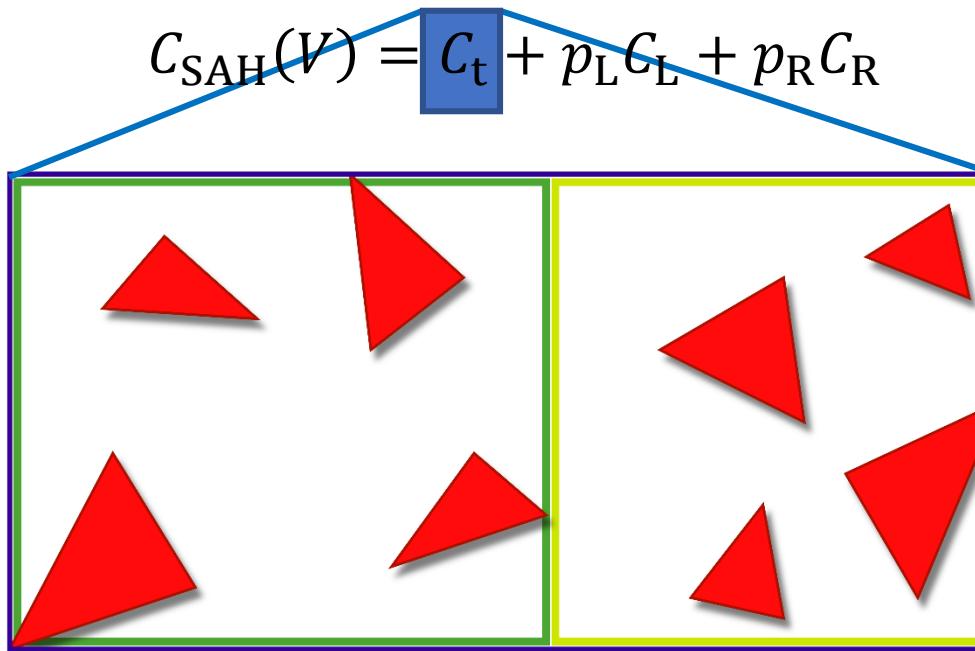


Surface Area Heuristic

Where to put the splitting plane?

→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + p_L C_L + p_R C_R$$

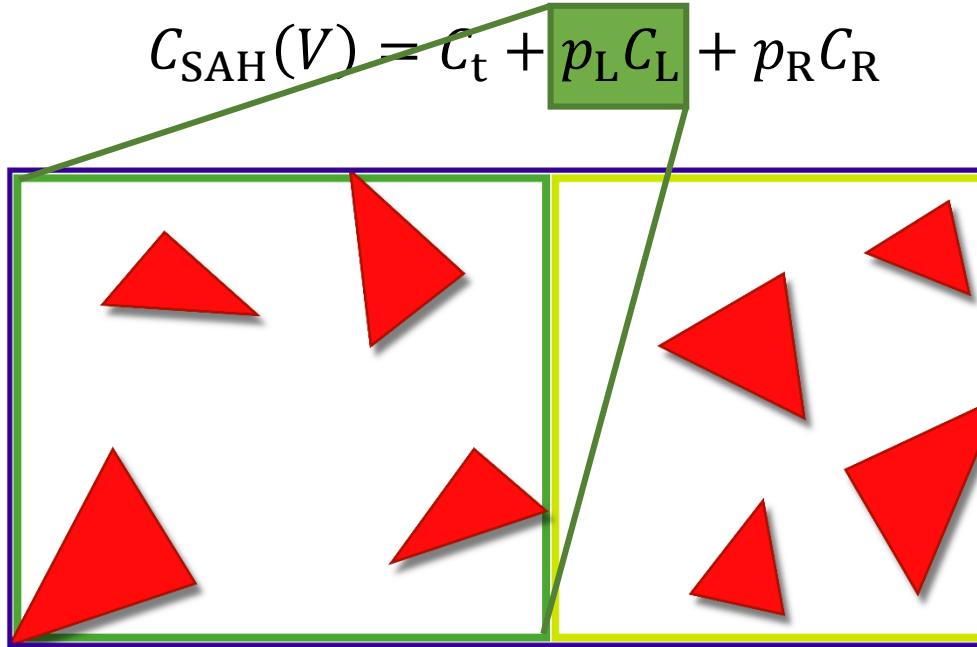


Surface Area Heuristic

Where to put the splitting plane?

→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + p_L C_L + p_R C_R$$

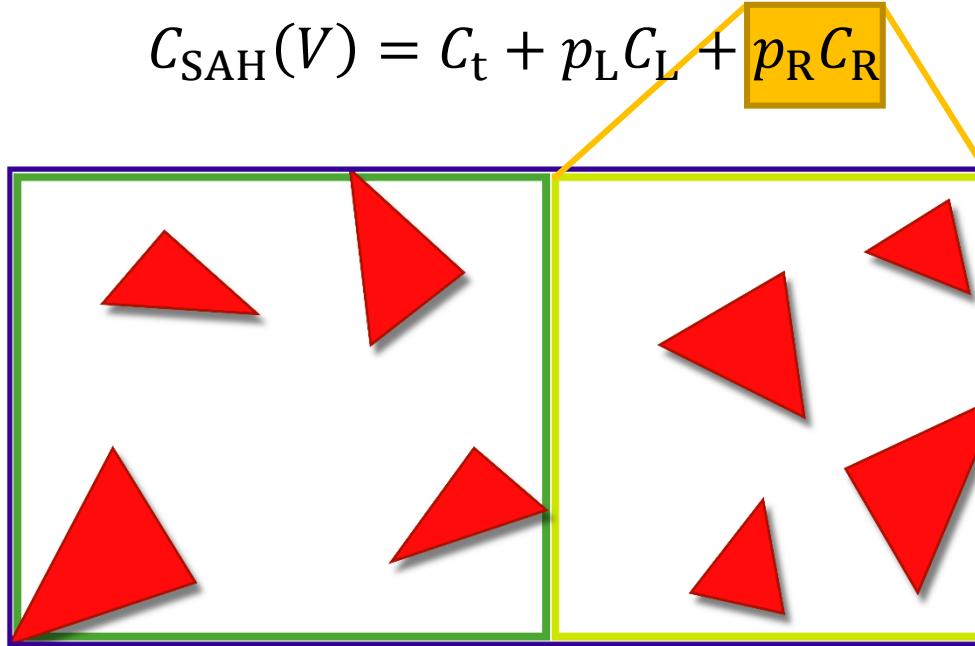


Surface Area Heuristic

Where to put the splitting plane?

→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + p_L C_L + p_R C_R$$

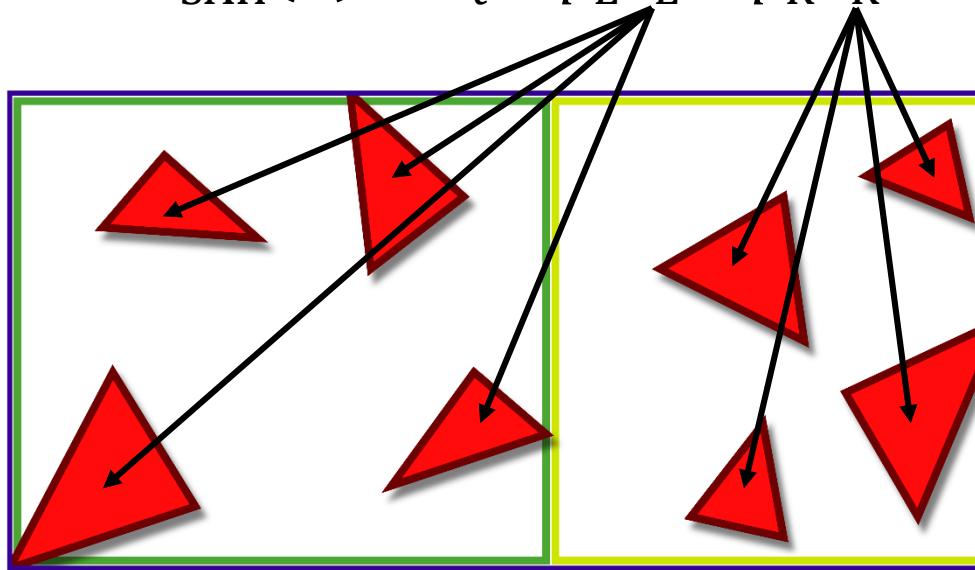


Surface Area Heuristic

Where to put the splitting plane?

→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + p_L C_L + p_R C_R$$

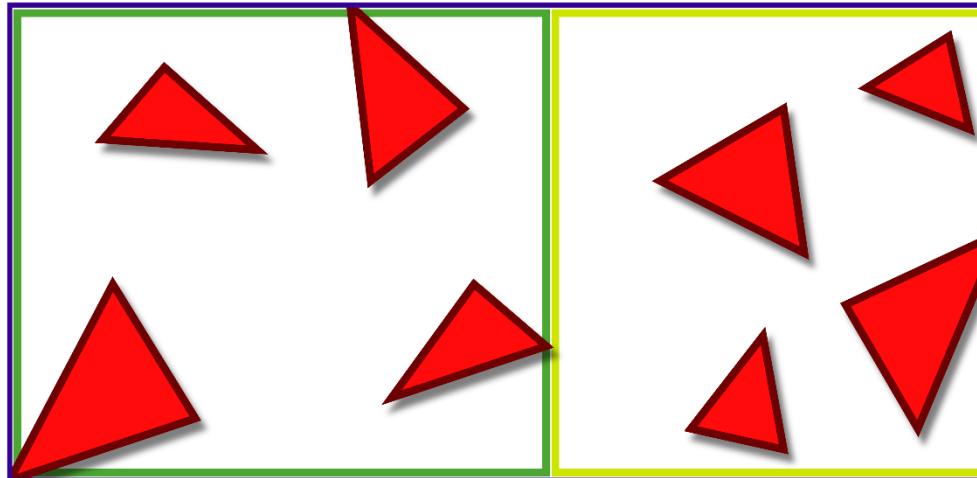


Surface Area Heuristic

Where to put the splitting plane?

→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + p_L C_L + p_R C_R$$

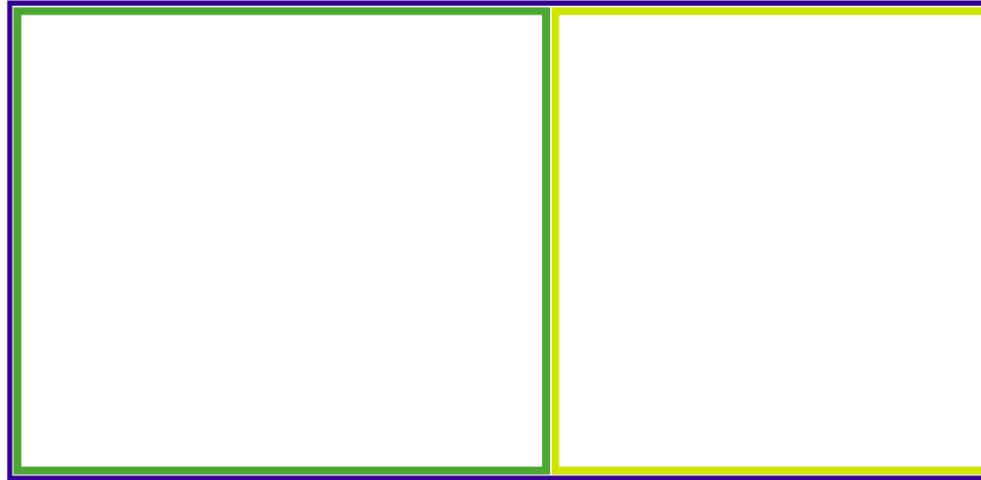


Surface Area Heuristic

Where to put the splitting plane?

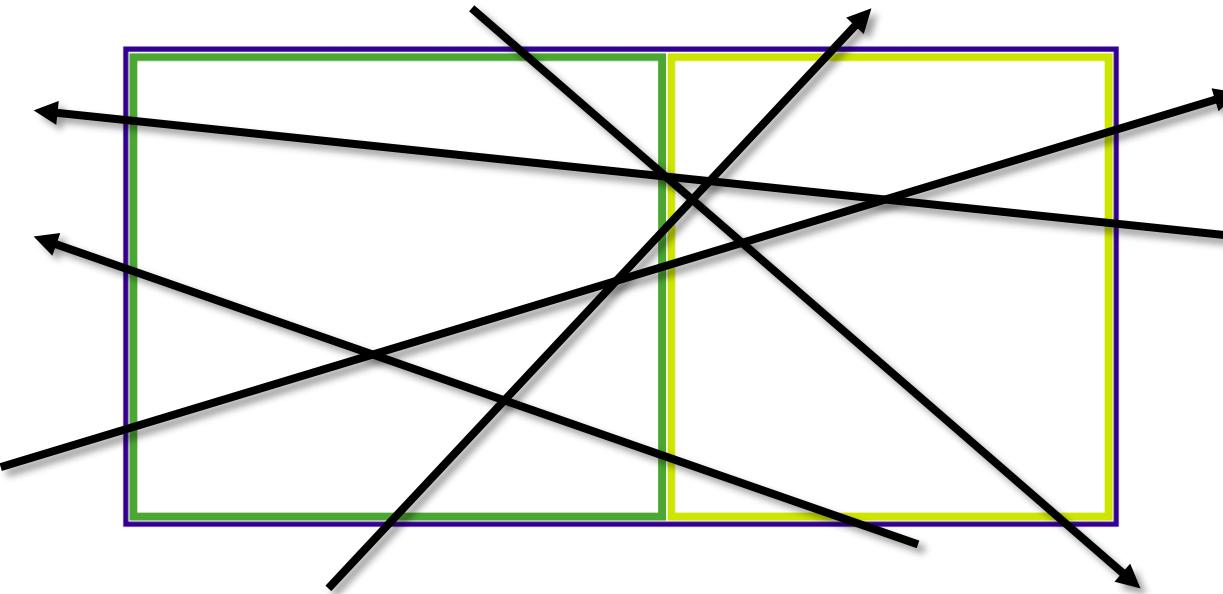
→ Try to minimize the ray tracing cost!

$$C_{SAH}(V) = C_t + \frac{SA_L}{SA_{Tot}} C_L + \frac{SA_R}{SA_{Tot}} C_R$$



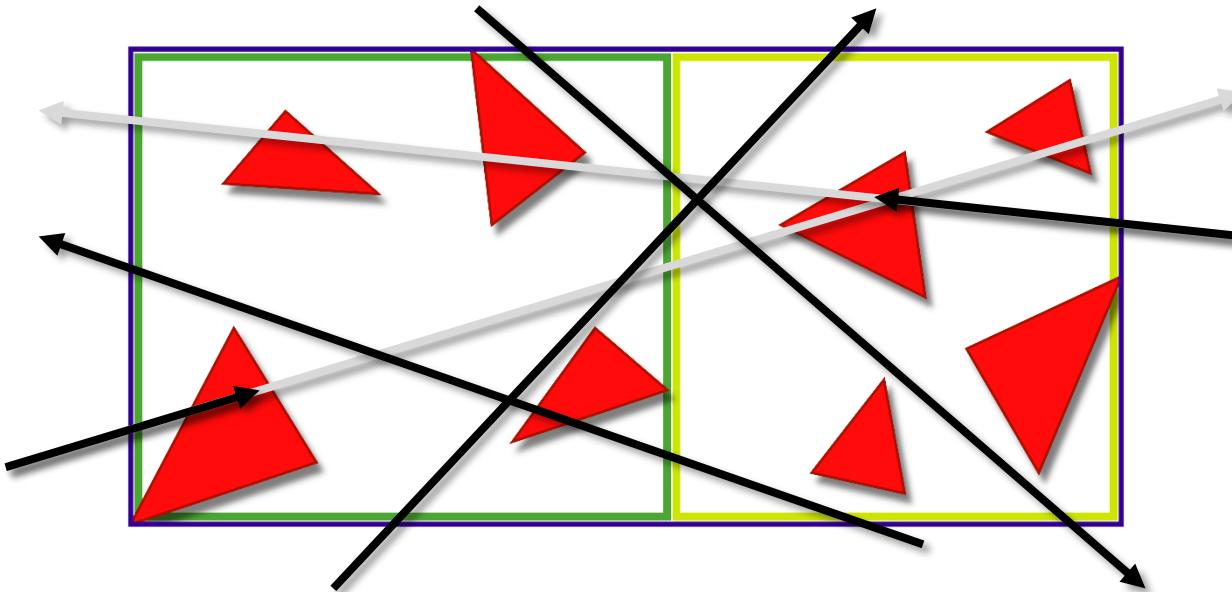
Surface Area Heuristic — Problems

SAH assumes that rays do not terminate within a voxel



Surface Area Heuristic — Problems

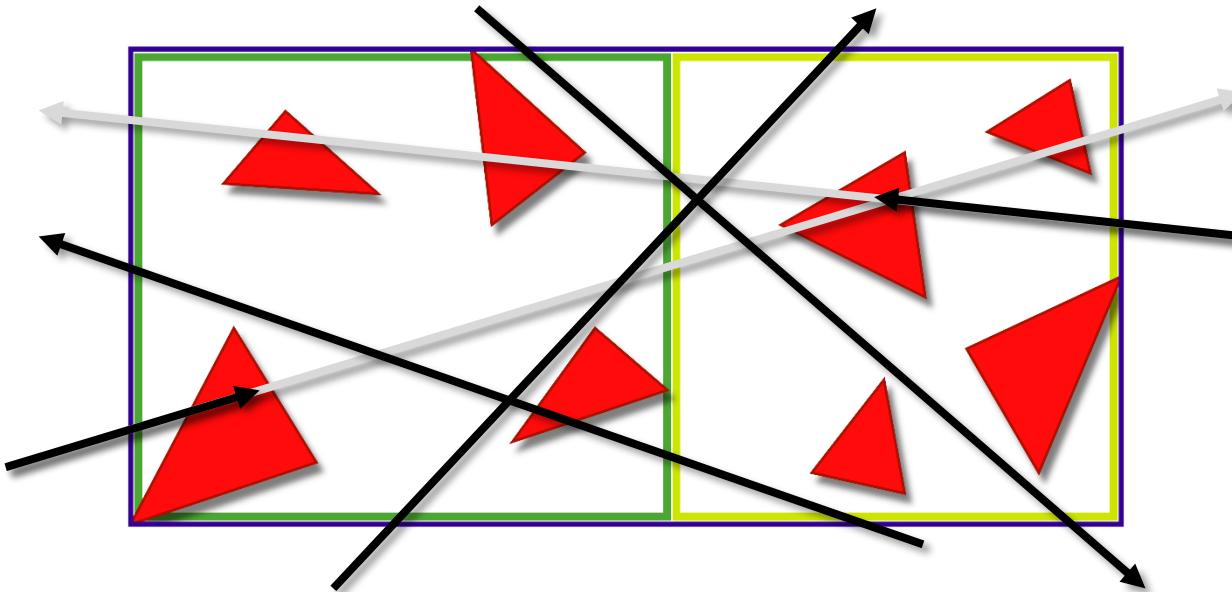
SAH assumes that rays do not terminate within a voxel



Surface Area Heuristic — Problems

SAH assumes that rays do not terminate within a voxel

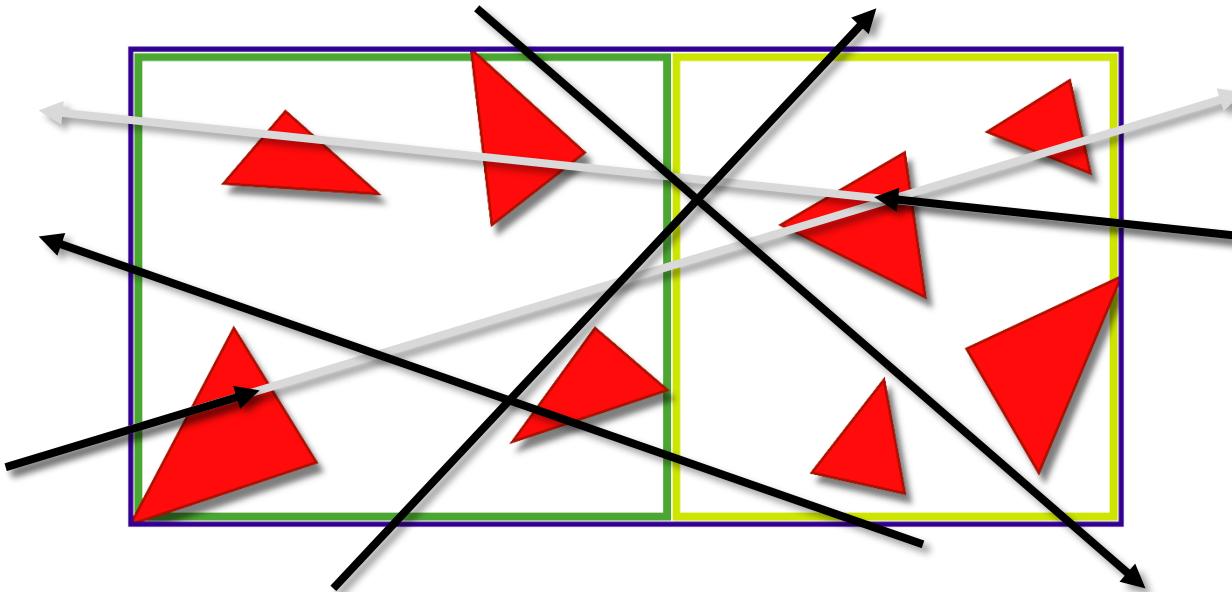
Contribution: $\mathcal{V}_{L \rightarrow R}$ = unblocked rays from left to right



Surface Area Heuristic — Problems

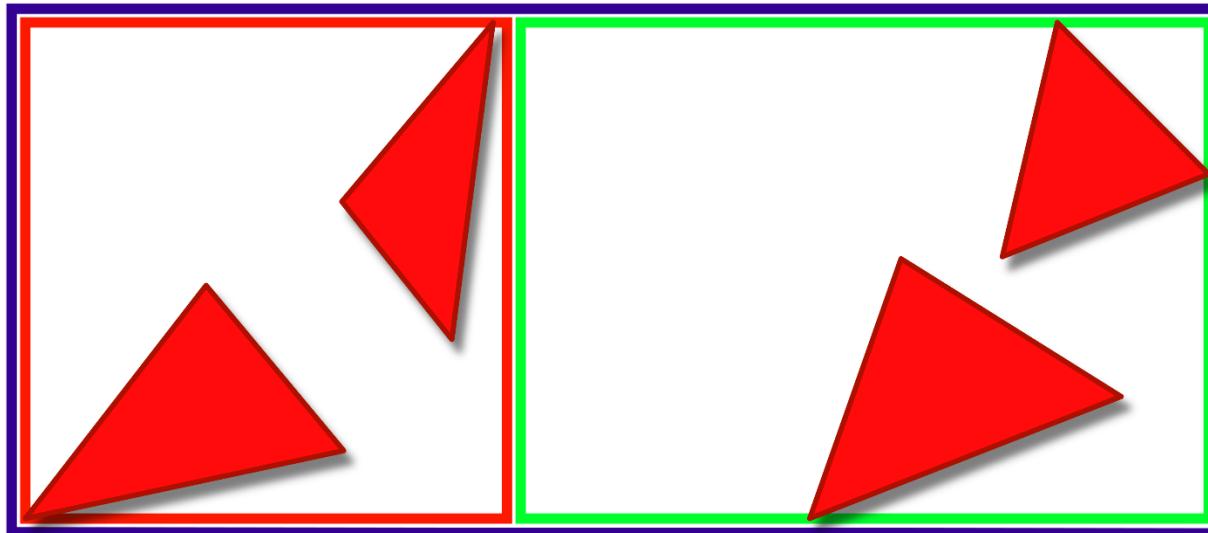
SAH assumes that rays do not terminate within a voxel

Contribution: $\mathcal{V}_{R \rightarrow L}$ = unblocked rays from right to left



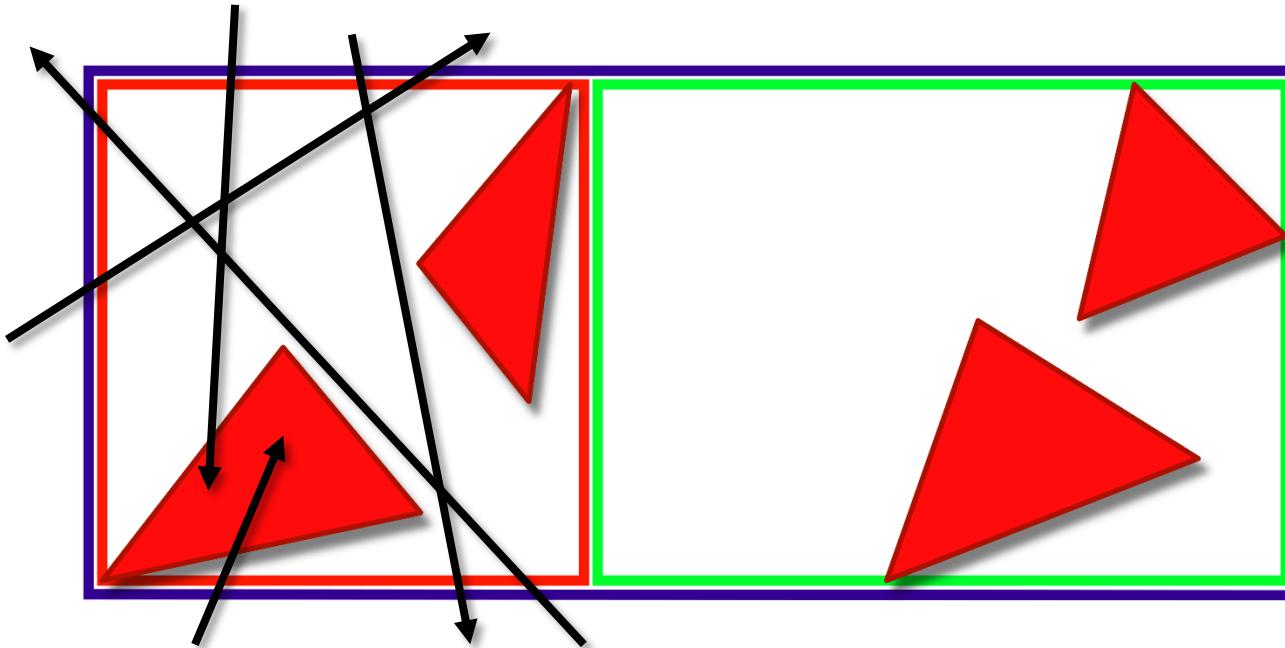
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL}C_L + p_{jR}C_R + \\ p_{L \rightarrow R}(C_L + \mathcal{V}_{L \rightarrow R}C_R) + p_{R \rightarrow L}(C_R + \mathcal{V}_{R \rightarrow L}C_L) \quad [\text{Ize et al. 2011}]$$



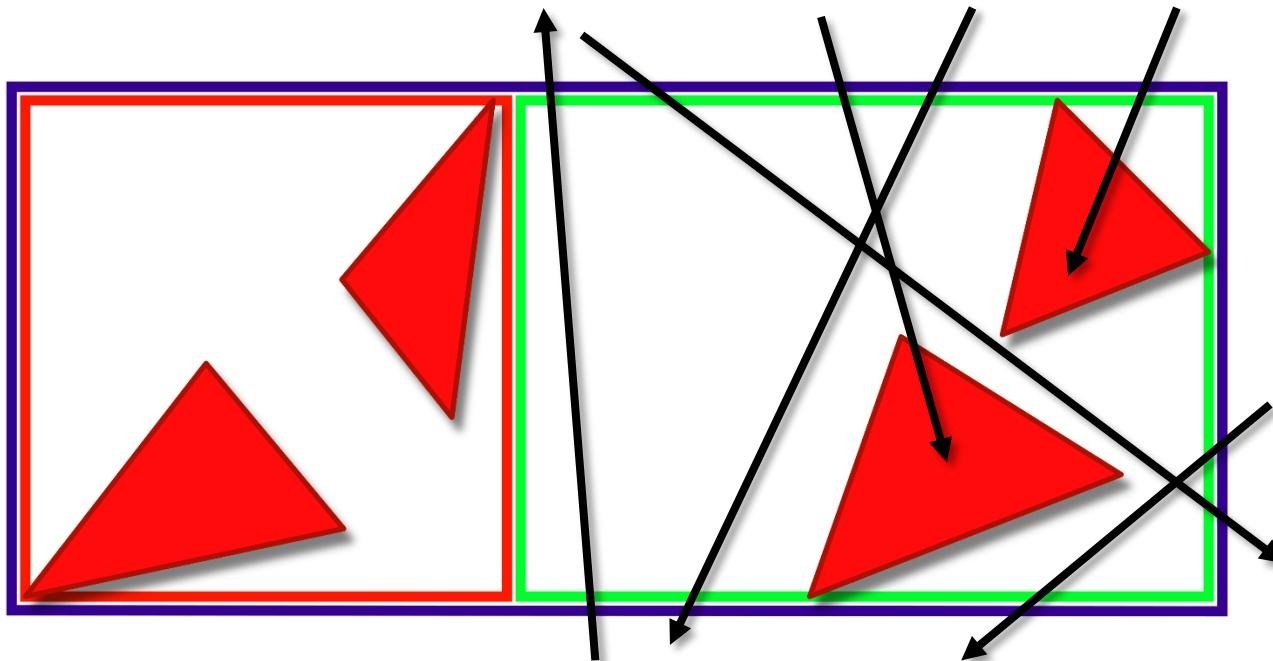
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL} C_L + p_{jR} C_R + \\ p_{L \rightarrow R} (C_L + \mathcal{V}_{L \rightarrow R} C_R) + p_{R \rightarrow L} (C_R + \mathcal{V}_{R \rightarrow L} C_L)$$



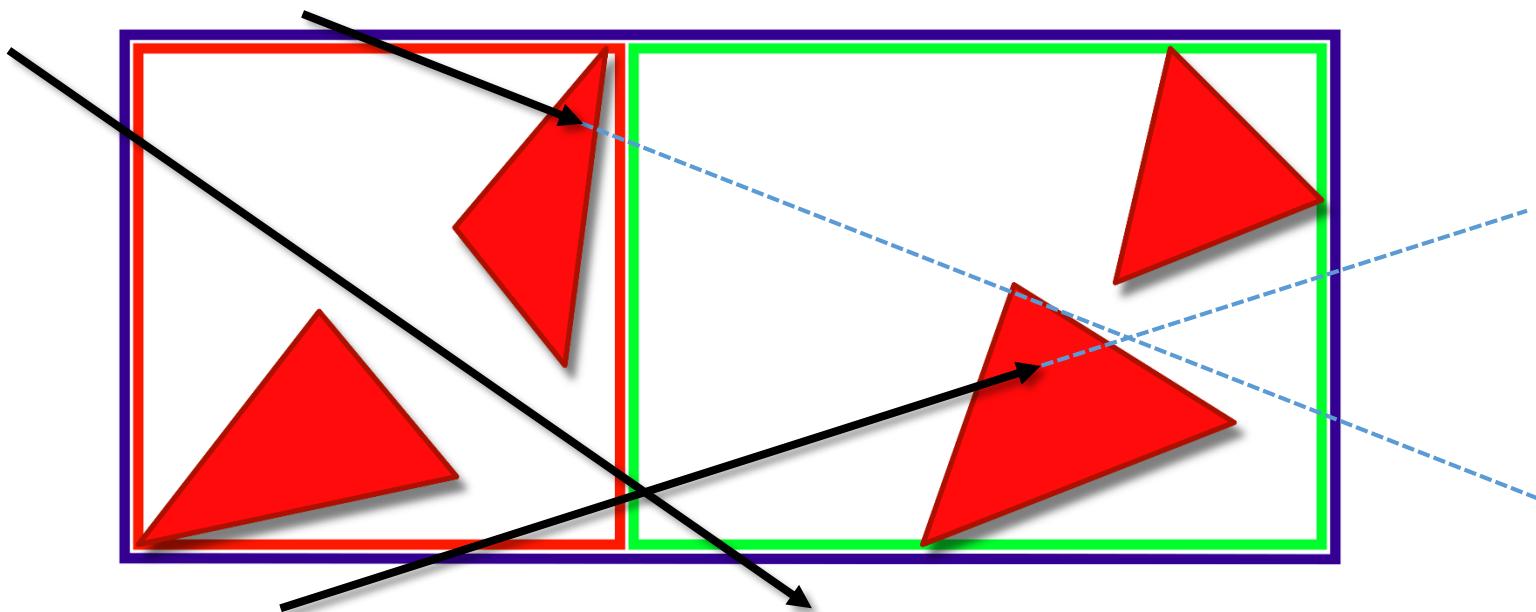
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL}C_L + p_{jR}C_R + \\ p_{L \rightarrow R}(C_L + \mathcal{V}_{L \rightarrow R}C_R) + p_{R \rightarrow L}(C_R + \mathcal{V}_{R \rightarrow L}C_L)$$



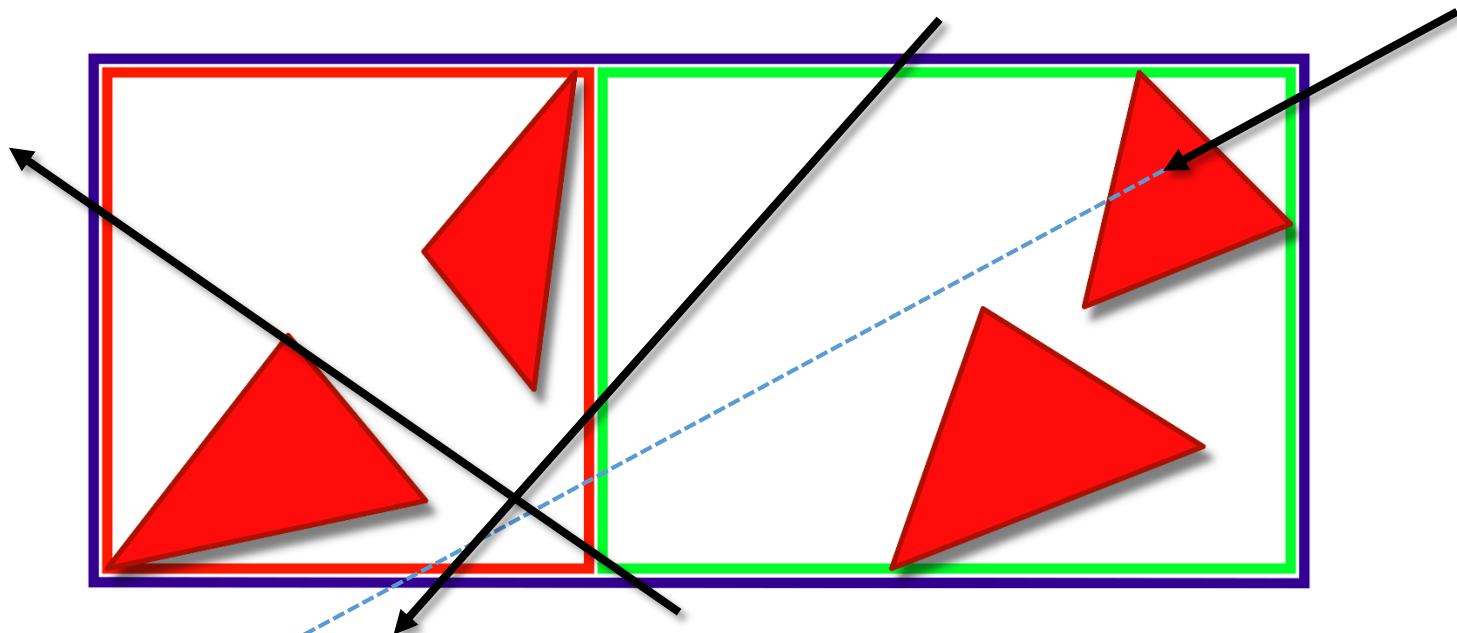
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL}C_L + p_{jR}C_R + \\ p_{L \rightarrow R}(C_L + \mathcal{V}_{L \rightarrow R}C_R) + p_{R \rightarrow L}(C_R + \mathcal{V}_{R \rightarrow L}C_L)$$



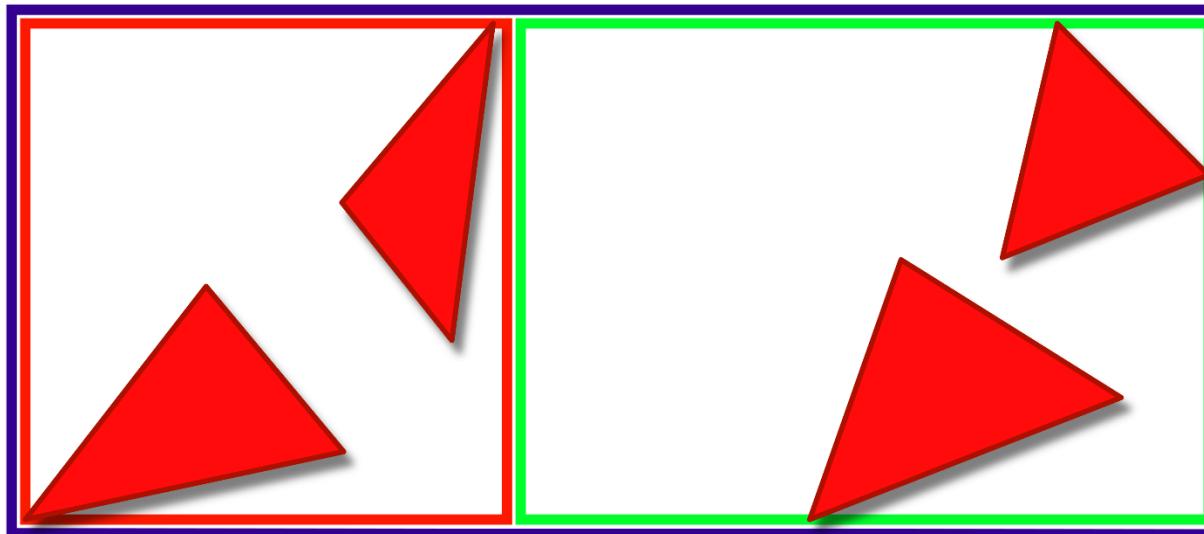
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL}C_L + p_{jR}C_R + \\ p_{L \rightarrow R}(C_L + \mathcal{V}_{L \rightarrow R}C_R) + p_{R \rightarrow L}(C_R + \mathcal{V}_{R \rightarrow L}C_L)$$



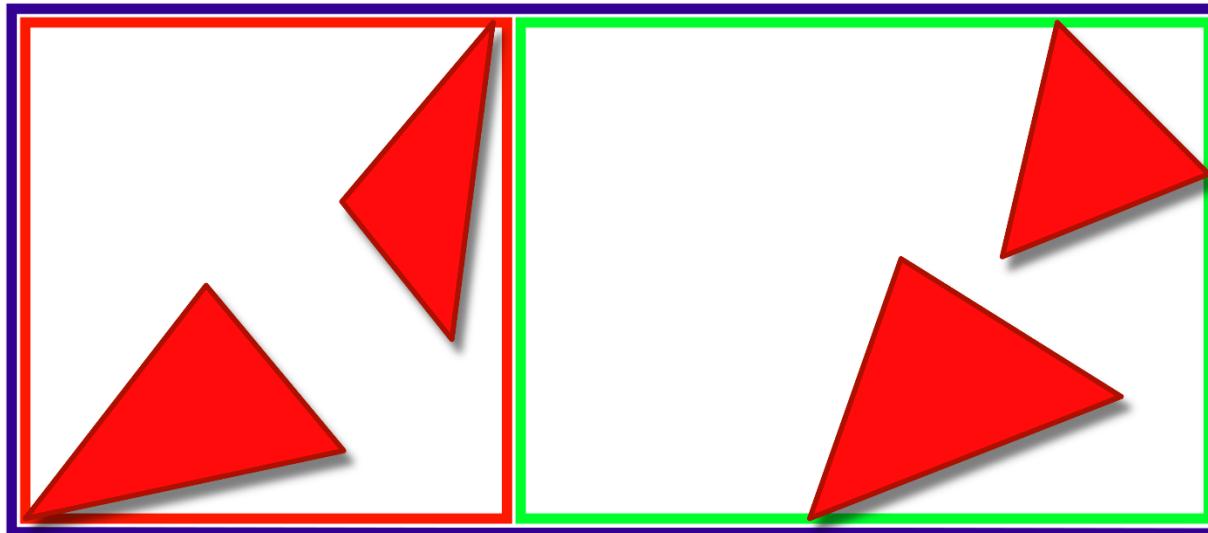
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL} C_L + p_{jR} C_R + \\ p_{L \rightarrow R} (C_L + \mathcal{V}_{L \rightarrow R} C_R) + p_{R \rightarrow L} (C_R + \mathcal{V}_{R \rightarrow L} C_L)$$



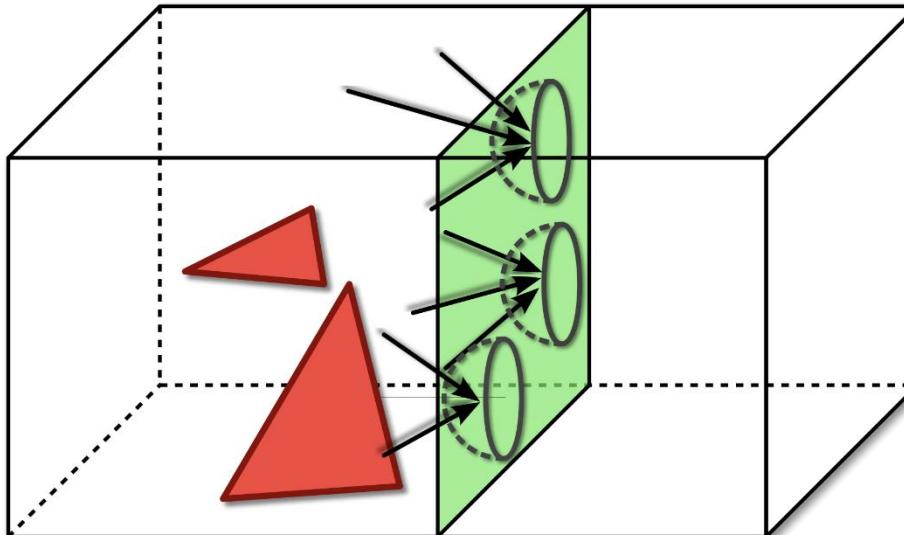
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL}C_L + p_{jR}C_R + \\ p_{L \rightarrow R}(C_L + \mathcal{V}_{L \rightarrow R}C_R) + p_{R \rightarrow L}(C_R + \mathcal{V}_{R \rightarrow L}C_L)$$



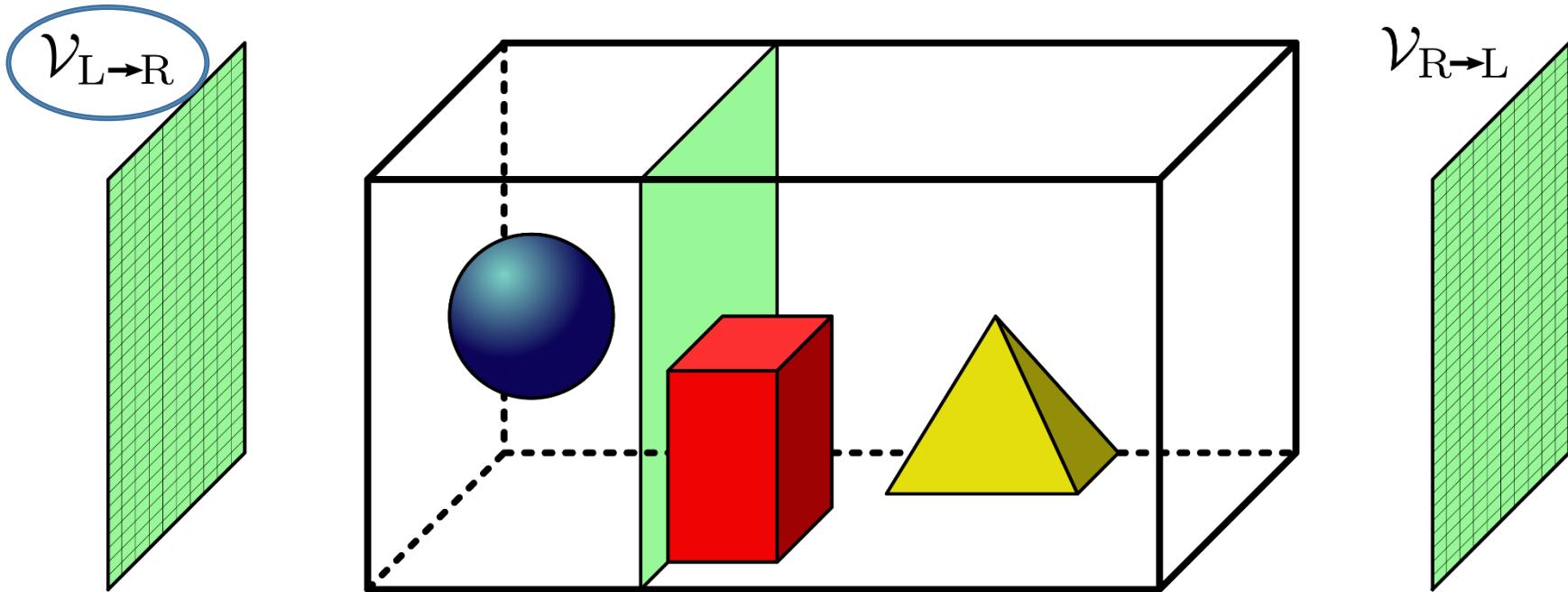
Ray Termination Surface Area Heuristic

$$C_{\text{RTSAH}} = C_t + p_{jL}C_L + p_{jR}C_R + \\ p_{L \rightarrow R}(C_L + \mathcal{V}_{L \rightarrow R}C_R) + p_{R \rightarrow L}(C_R + \mathcal{V}_{R \rightarrow L}C_L)$$



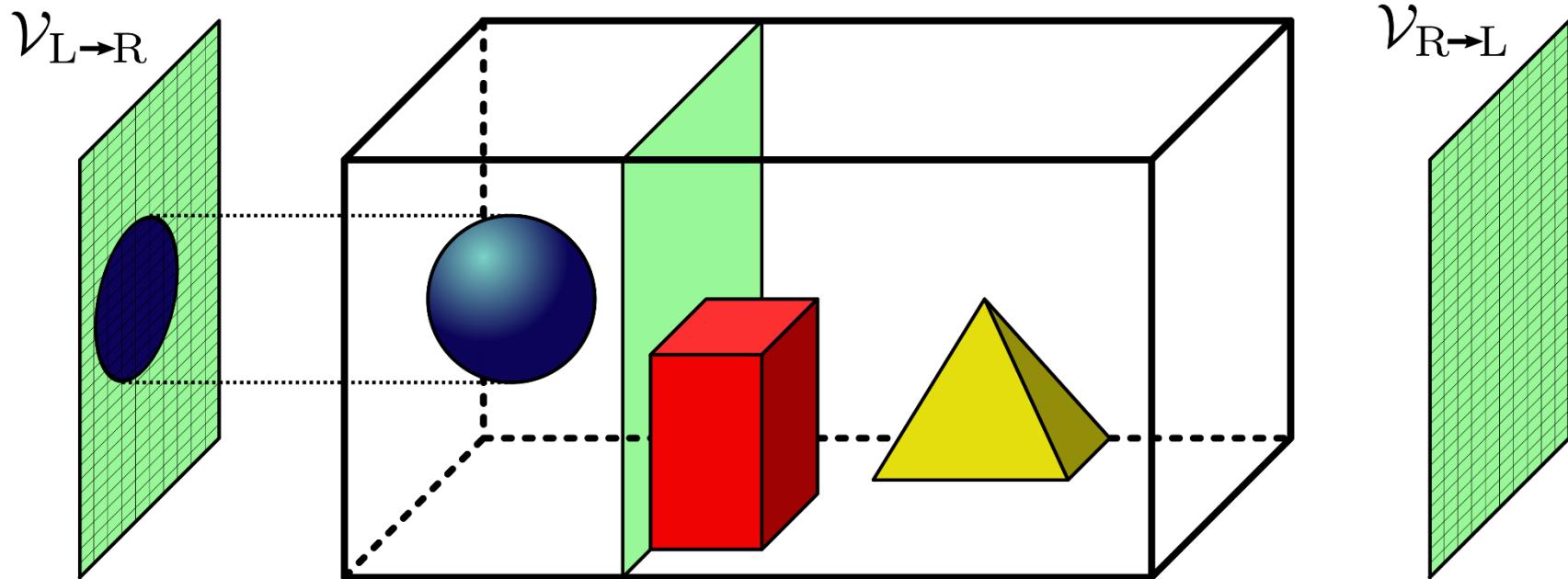
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



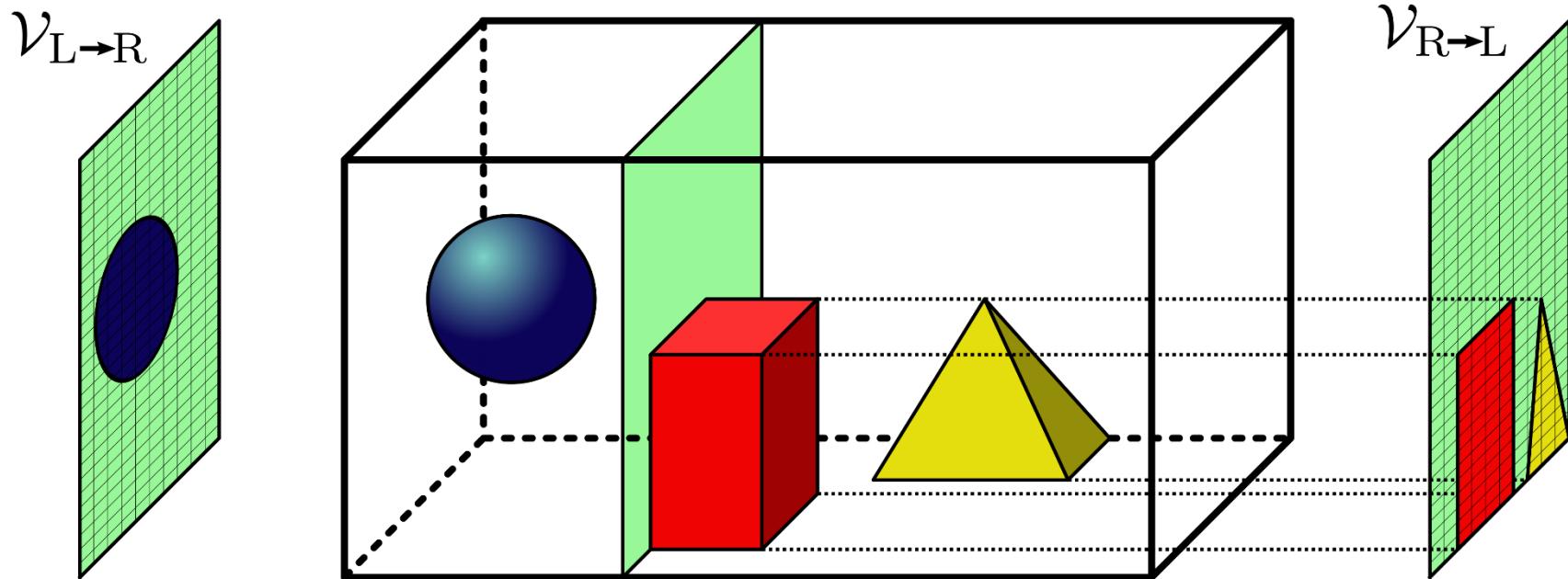
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



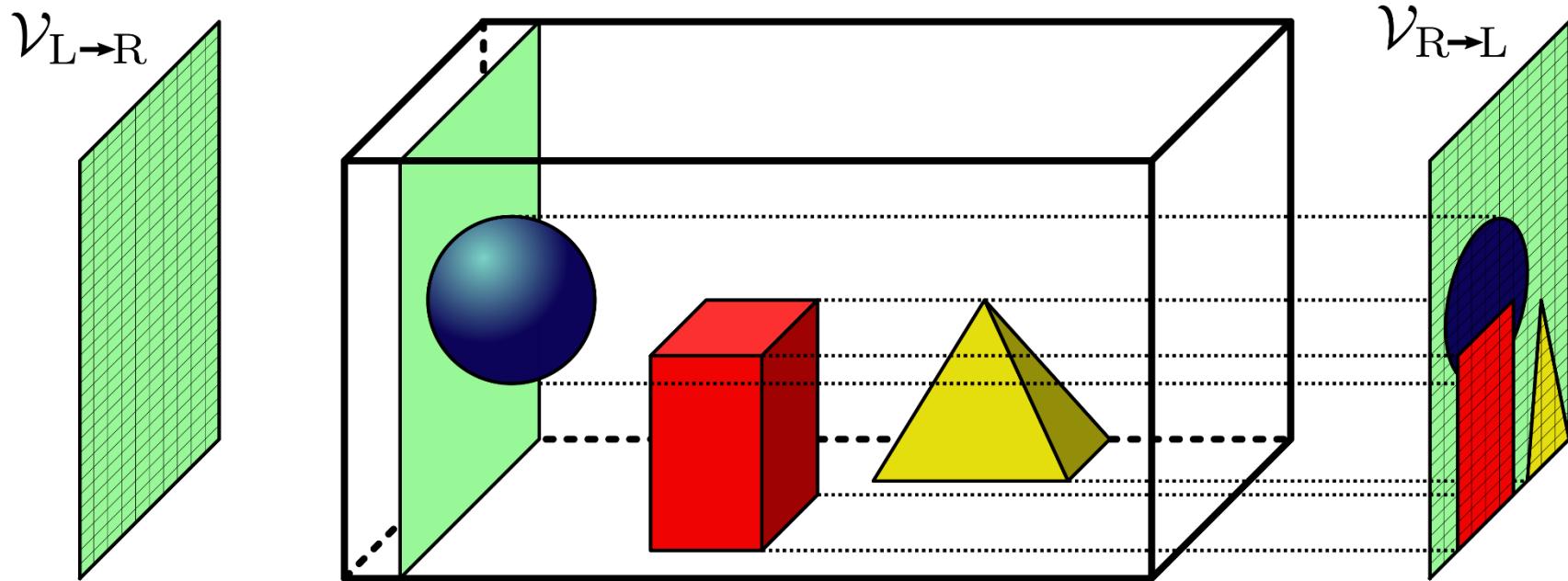
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



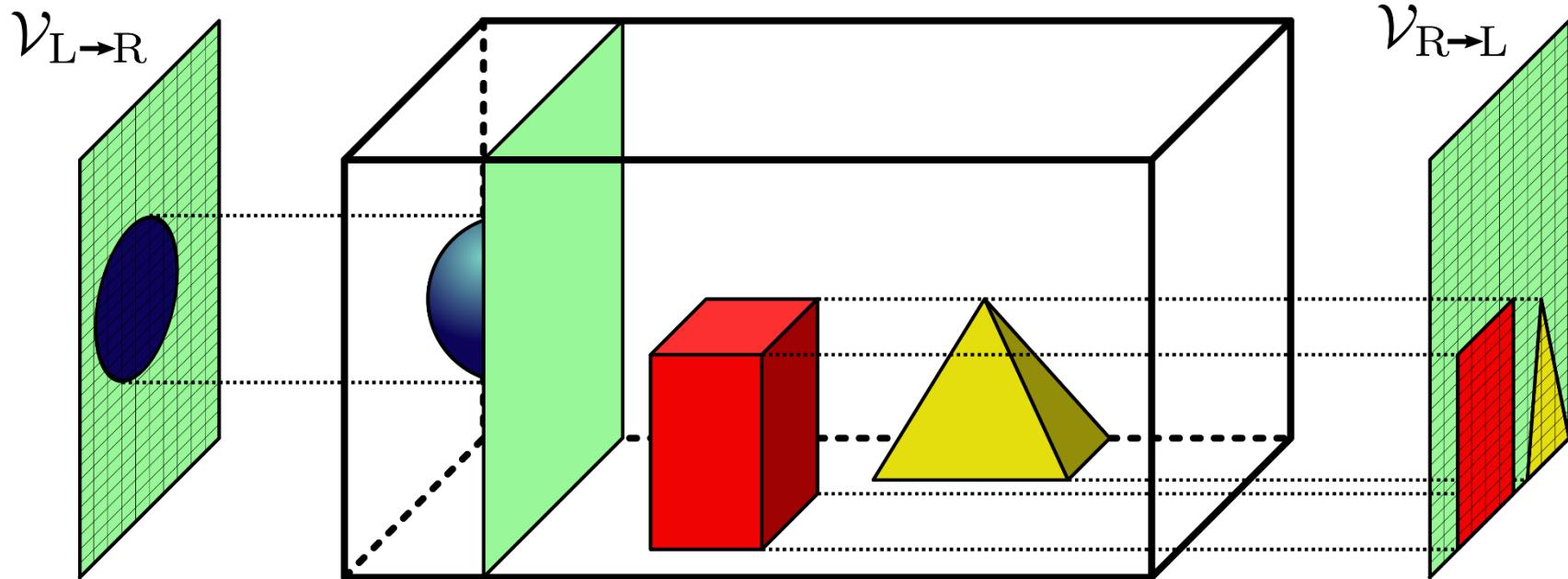
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



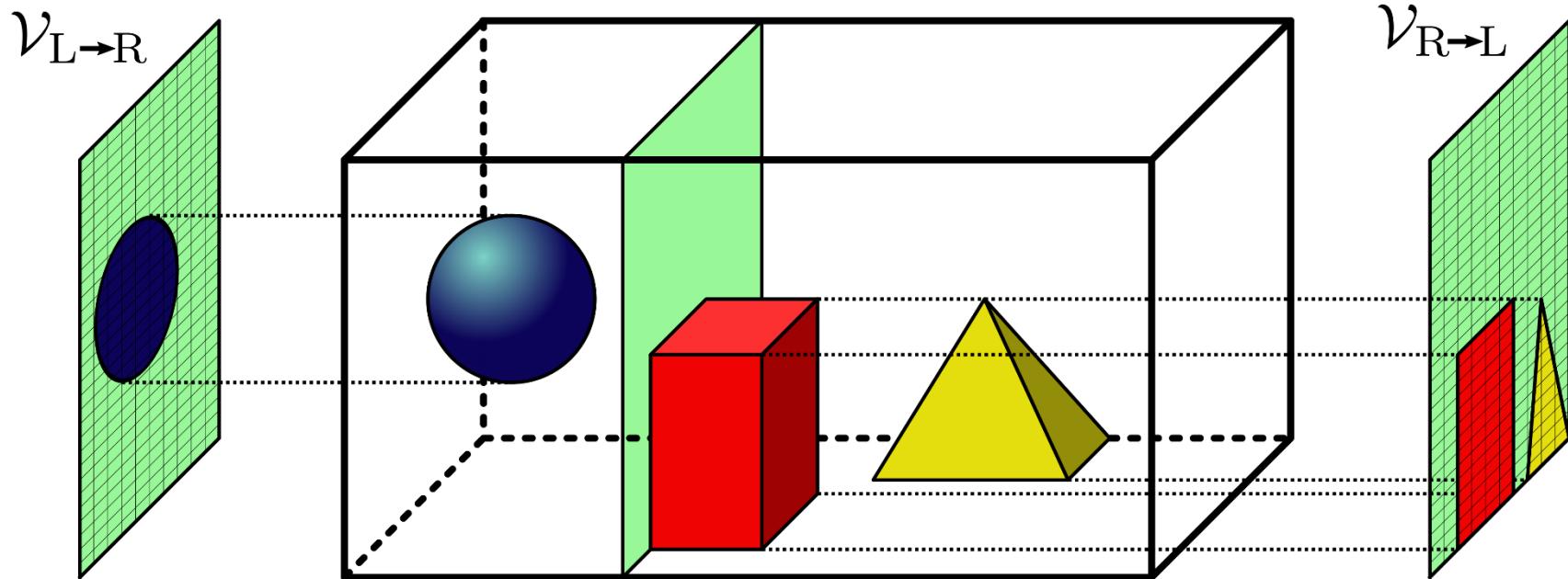
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



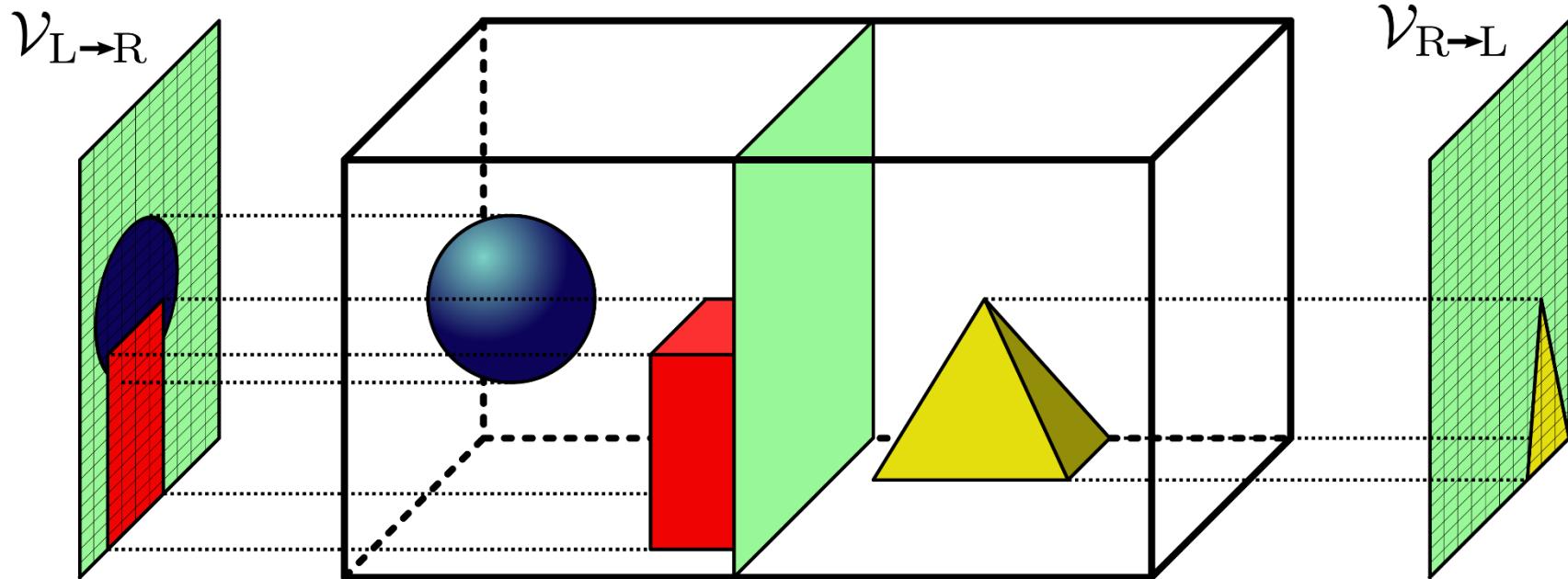
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



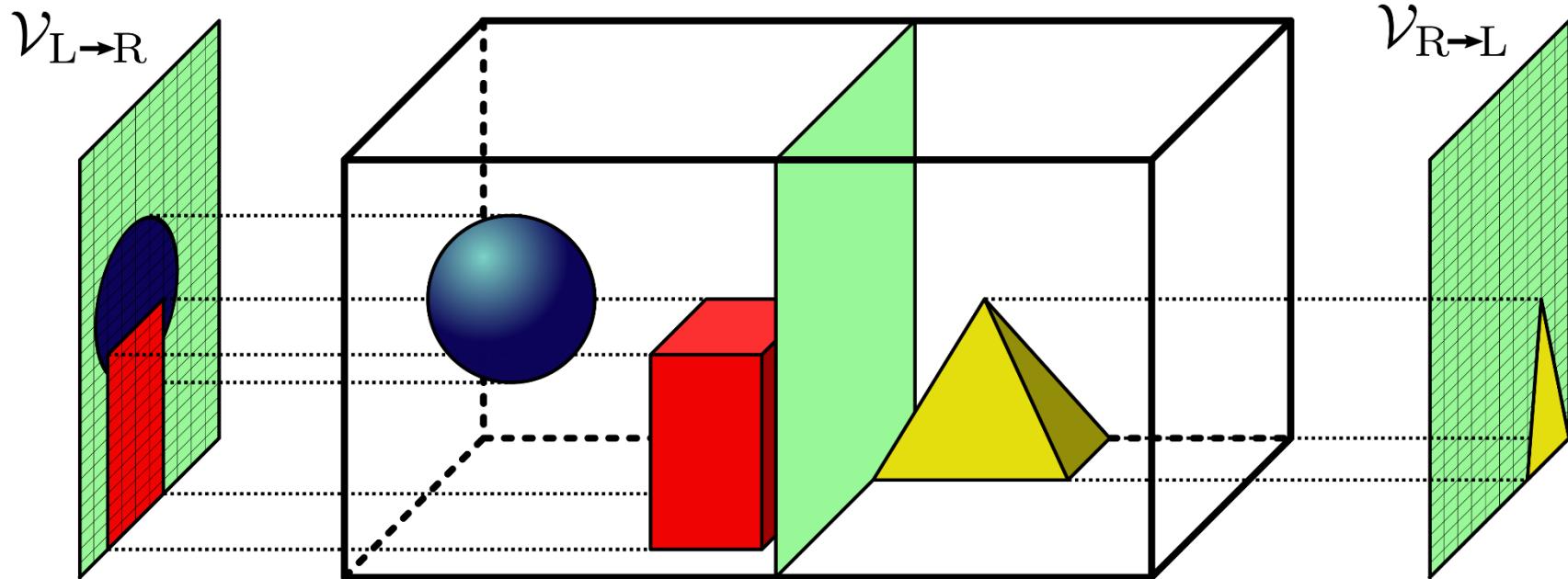
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane



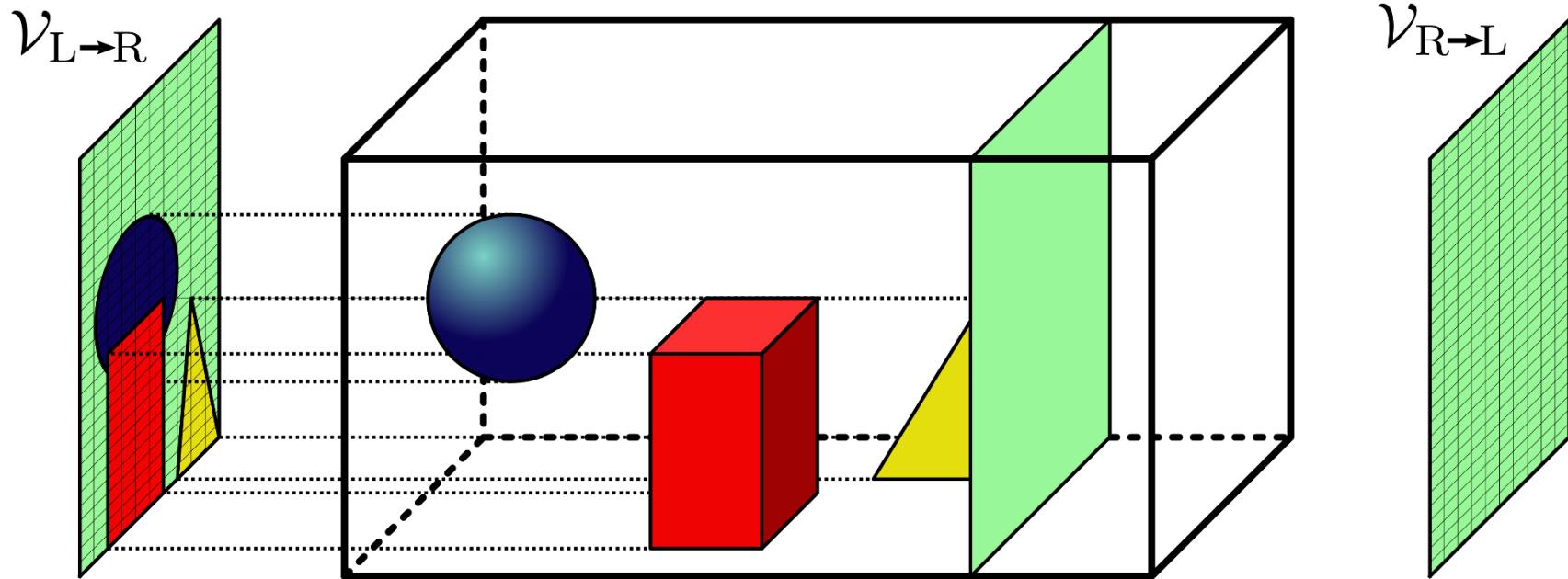
RASTERIZED Approximation

Projects all primitives orthogonally on the splitting plane

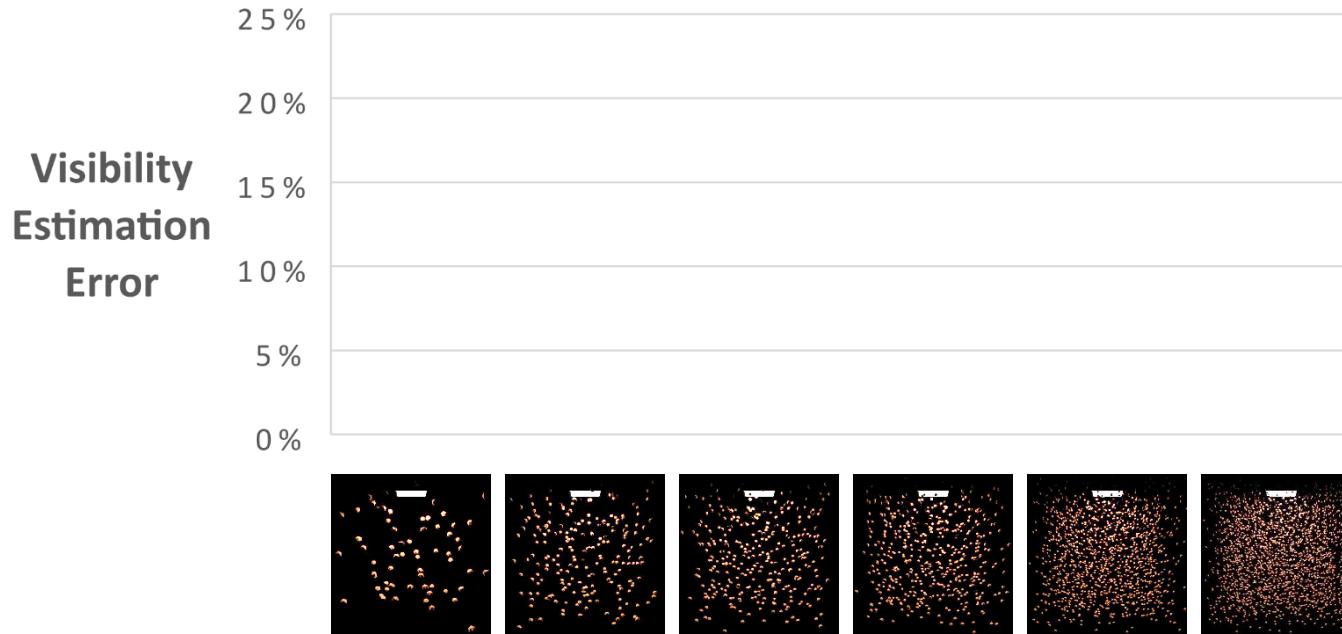


RASTERIZED Approximation

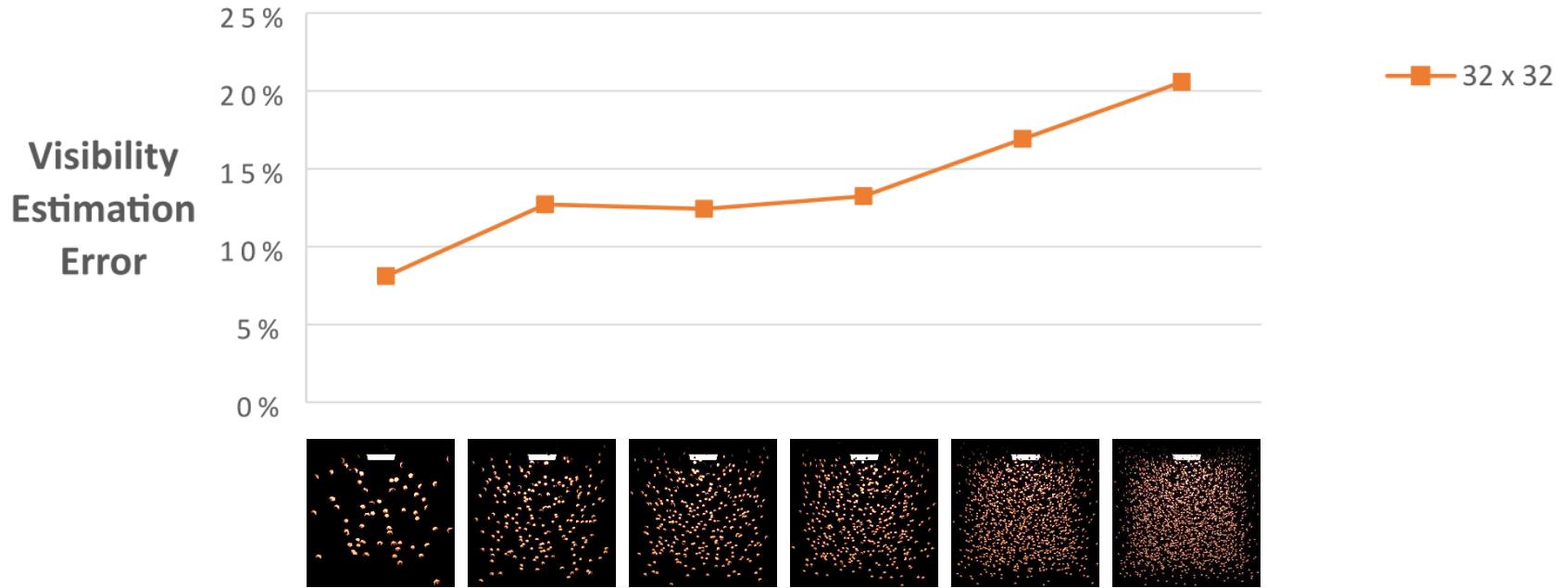
Projects all primitives orthogonally on the splitting plane



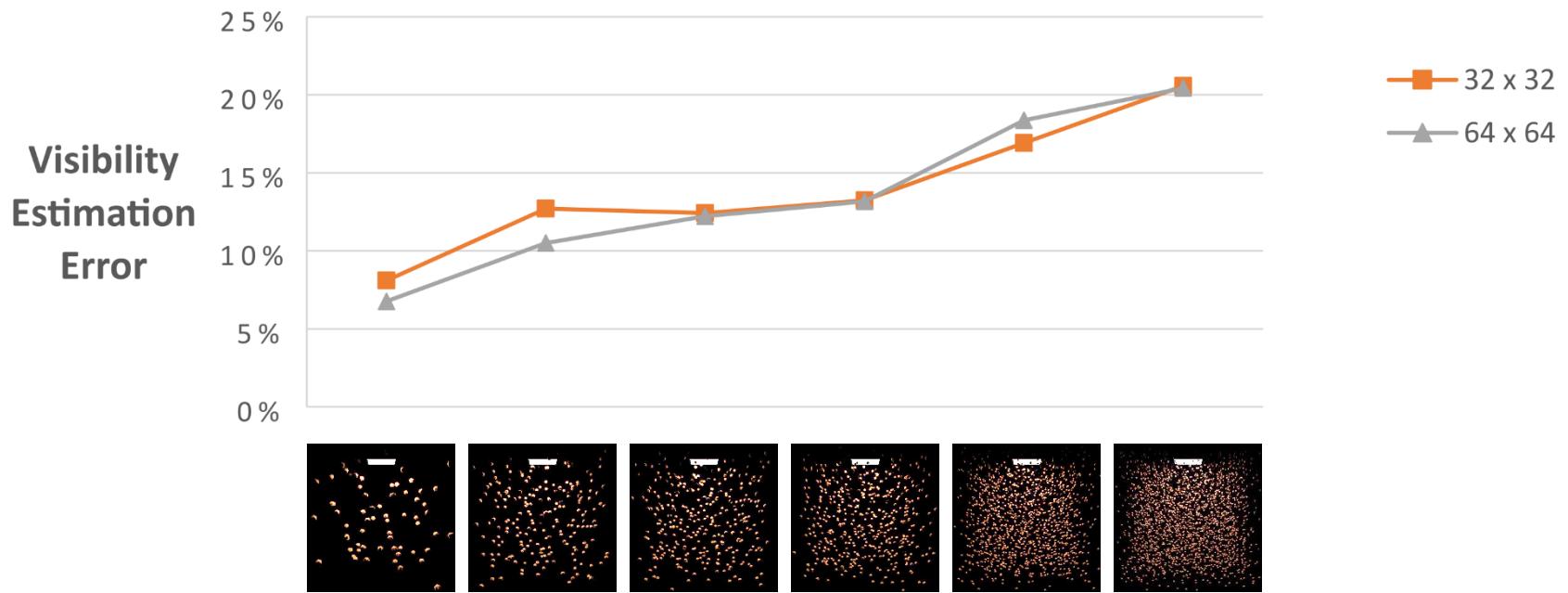
Visibility Estimation Error



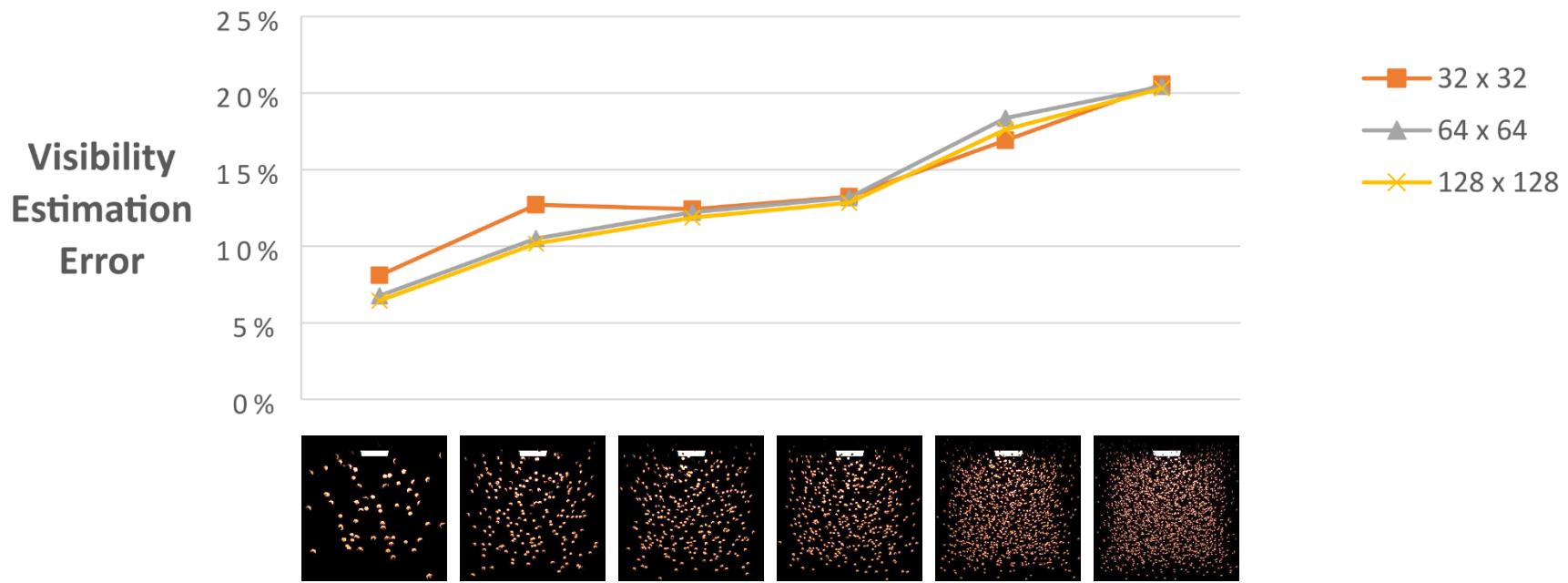
Visibility Estimation Error



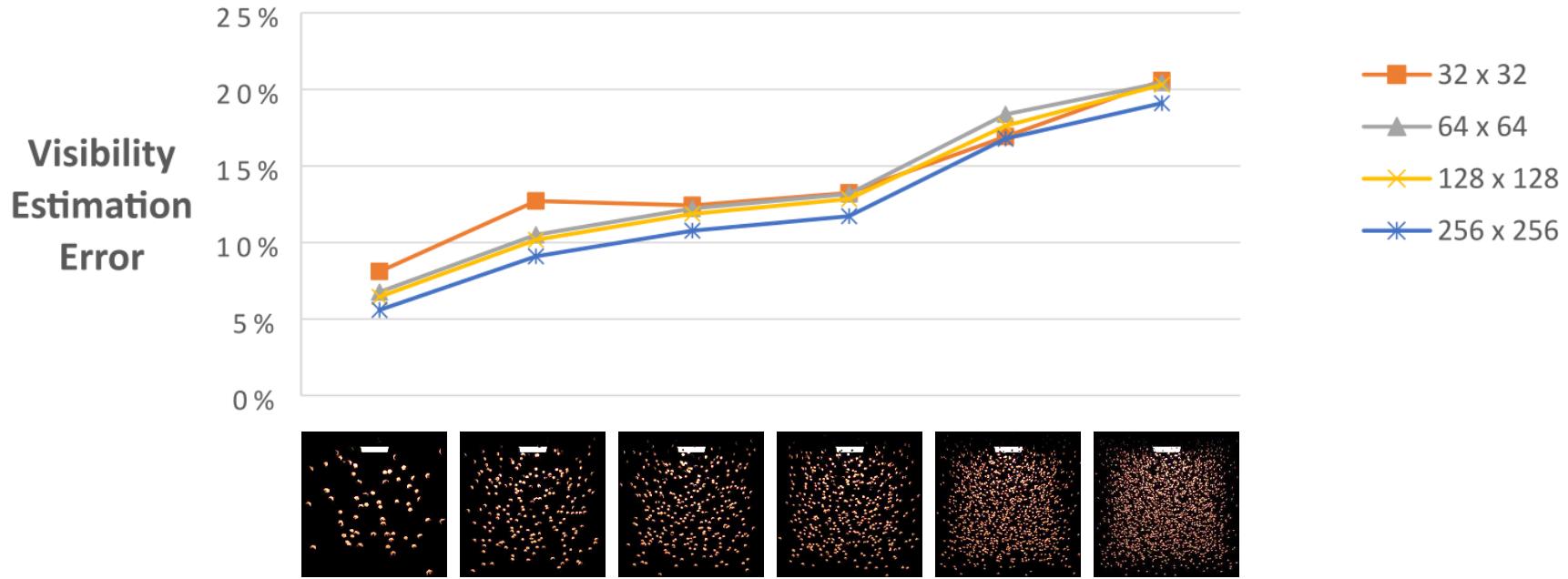
Visibility Estimation Error



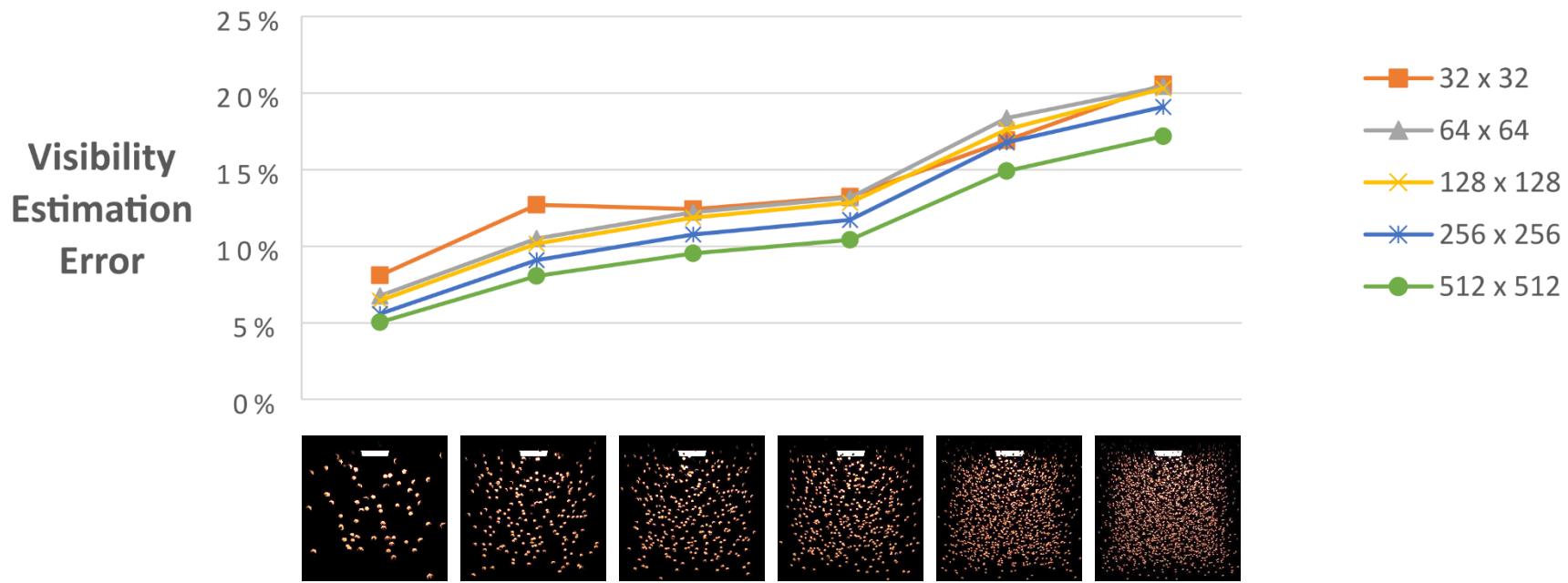
Visibility Estimation Error



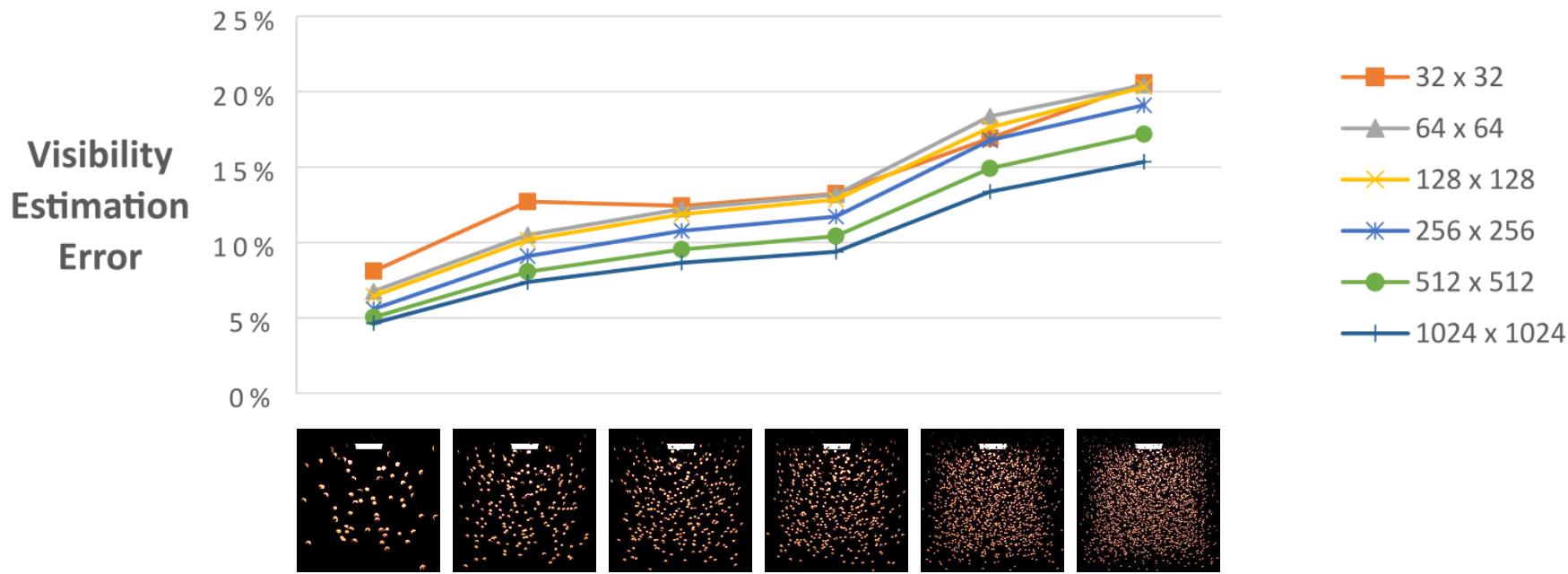
Visibility Estimation Error



Visibility Estimation Error

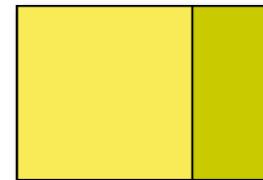
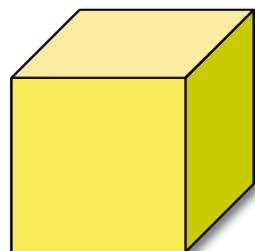


Visibility Estimation Error



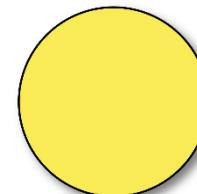
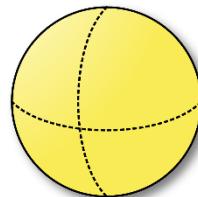
AVERAGED approximation

average projected surface area = $\frac{1}{4}$ surface area



$$SA = 6z^2$$

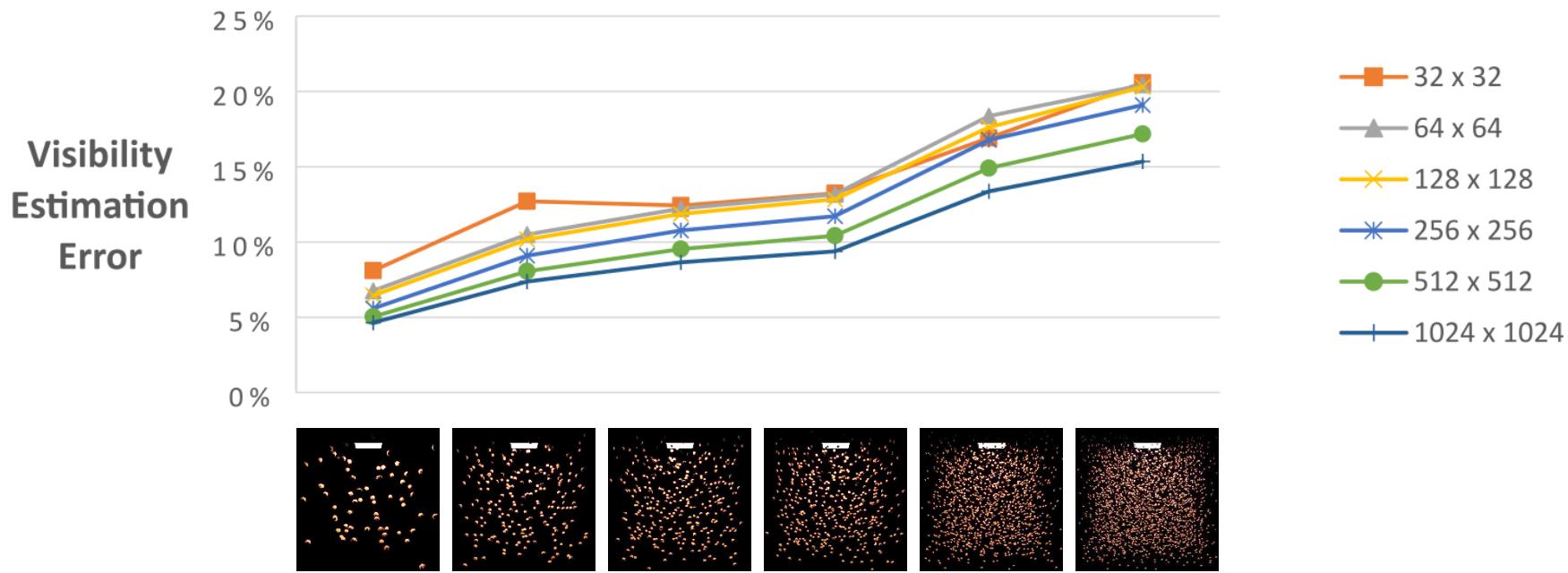
$$APSA = \frac{3}{2}z^2$$



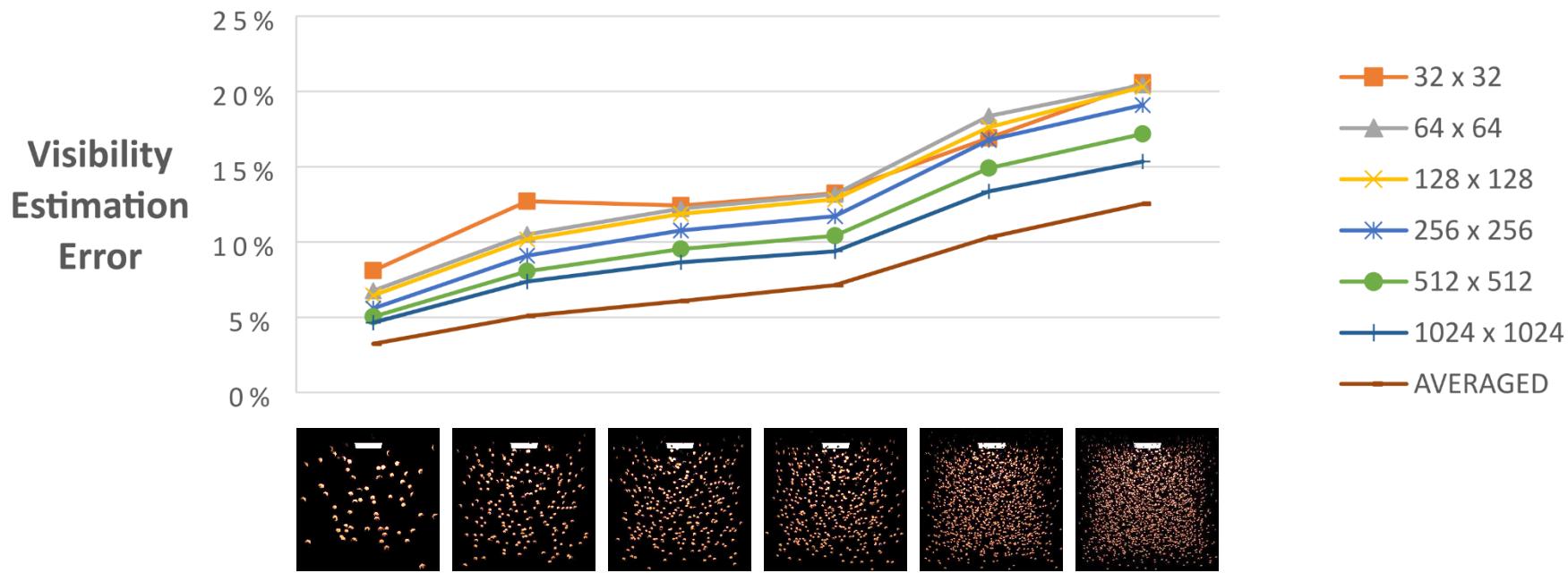
$$SA = 4\pi r^2$$

$$APSA = \pi r^2$$

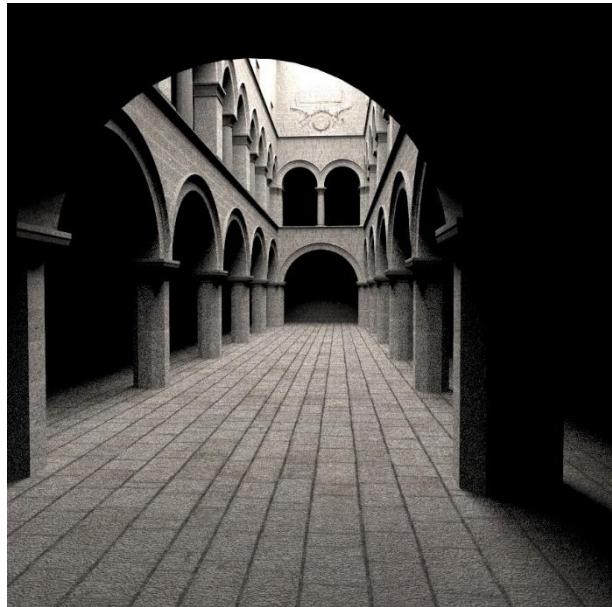
Visibility Estimation Error



Visibility Estimation Error

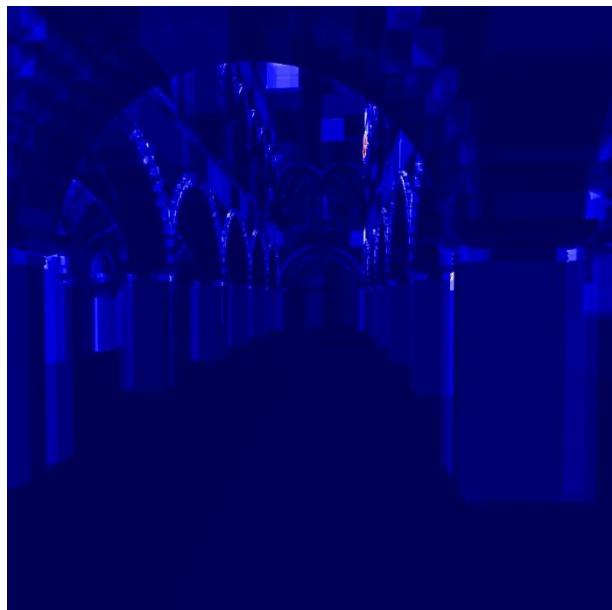


Results

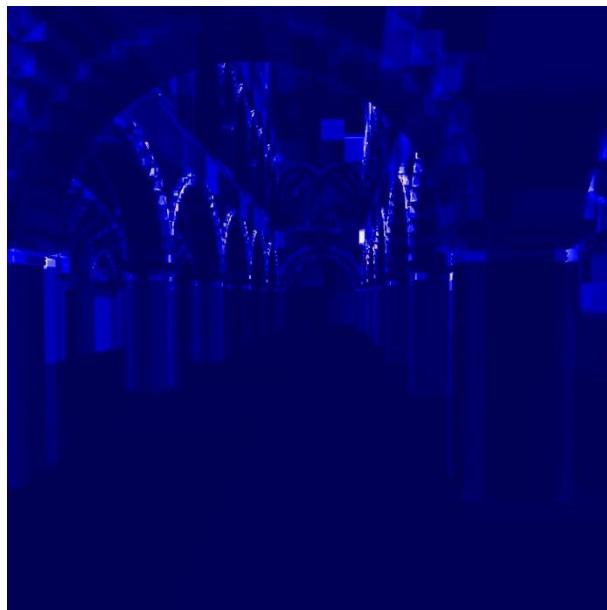


Sponza (512spp)
Whitted Ray Tracer

SAH



Diff(RASTERIZED, SAH)



Diff(AVERAGED, SAH)



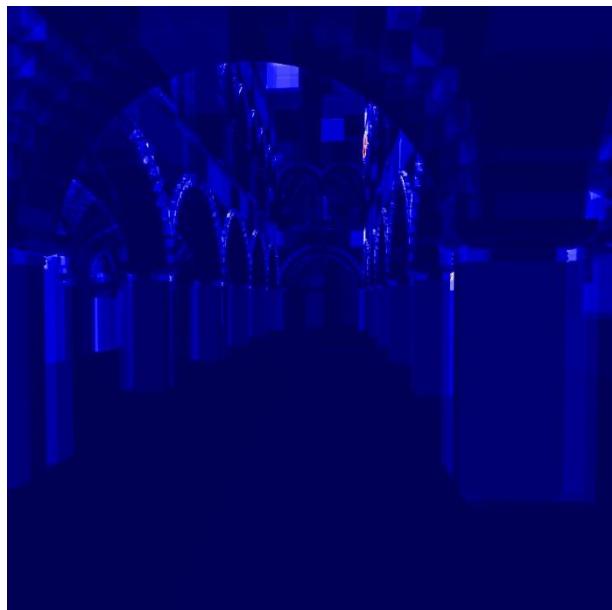
Low



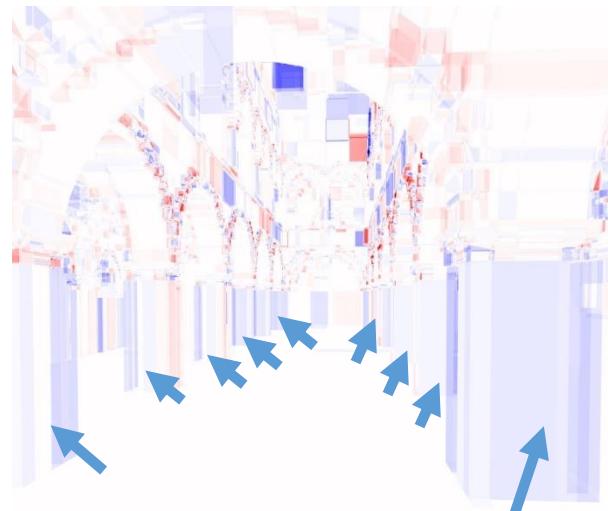
High

#INTERSECTION tests for PRIMARY rays

SAH



Diff(RASTERIZED, SAH)



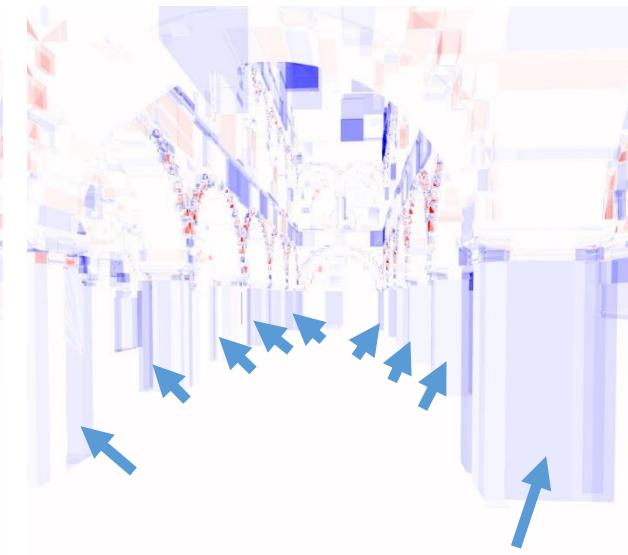
Less

-12.5%

More

-25.5%

Diff(AVERAGED, SAH)



#INTERSECTION tests for PRIMARY rays

SAH



Diff(RASTERIZED, SAH)



Diff(AVERAGED, SAH)



Low



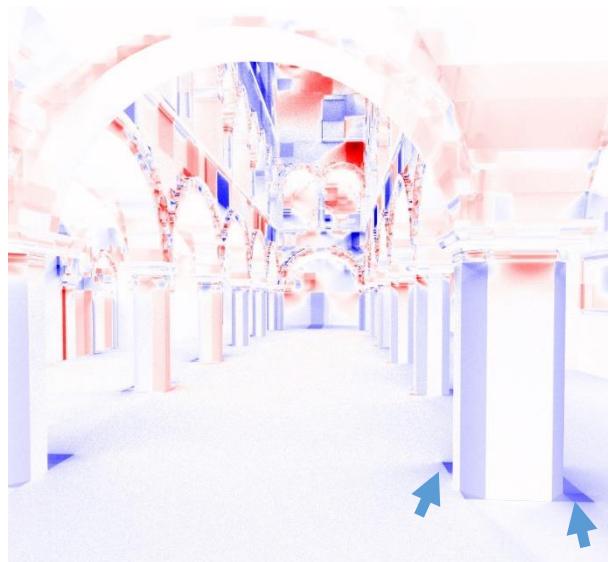
High

#INTERSECTION tests for SHADOW rays

SAH



Diff(RASTERIZED, SAH)



Less

-3.1%

More

-22.1%

#INTERSECTION tests for SHADOW rays

SAH



261.9s

Diff(RASTERIZED, SAH)



Less

251.6s (-3.9%)

Diff(AVERAGED, SAH)

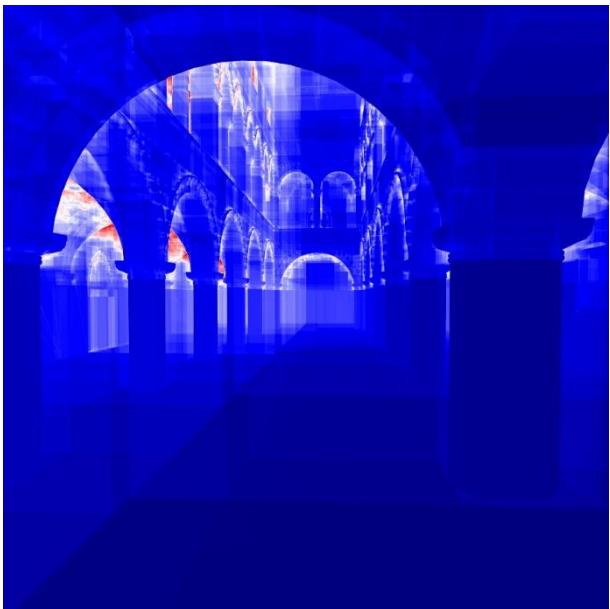


More

249.3s (-4.8%)

Total rendering time

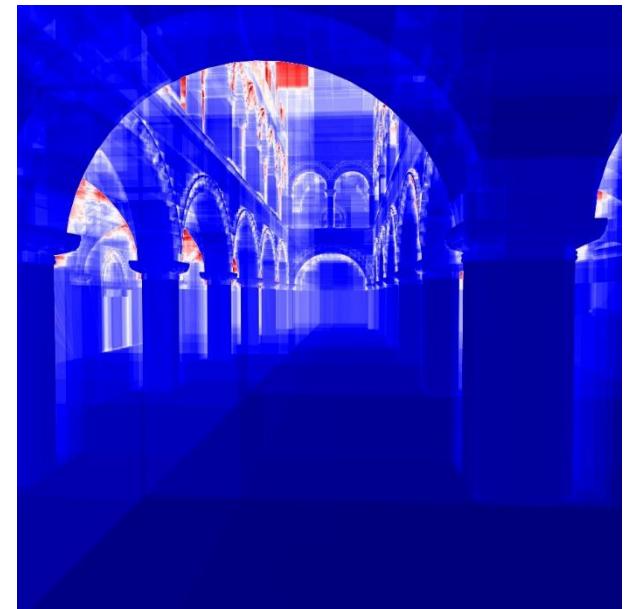
SAH



Diff(RASTERIZED, SAH)



Diff(AVERAGED, SAH)



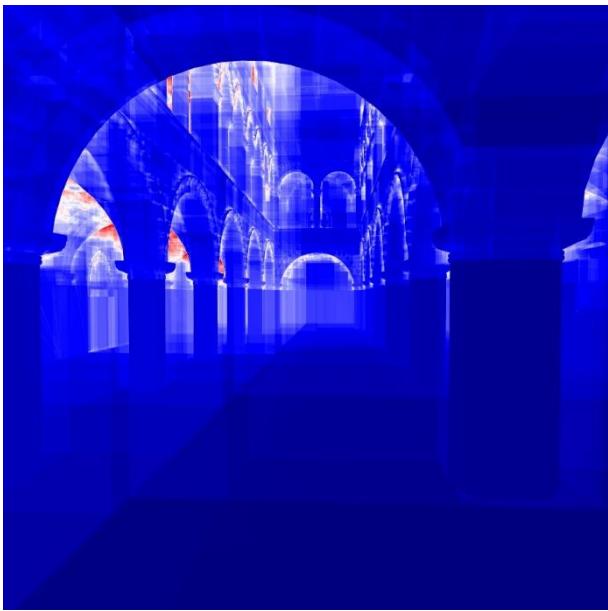
Low



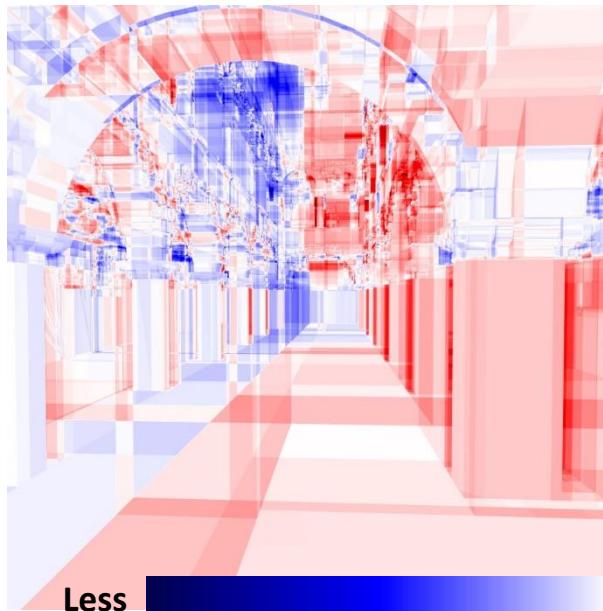
High

#TRAVESED nodes for PRIMARY rays

SAH



Diff(RASTERIZED, SAH)



Less

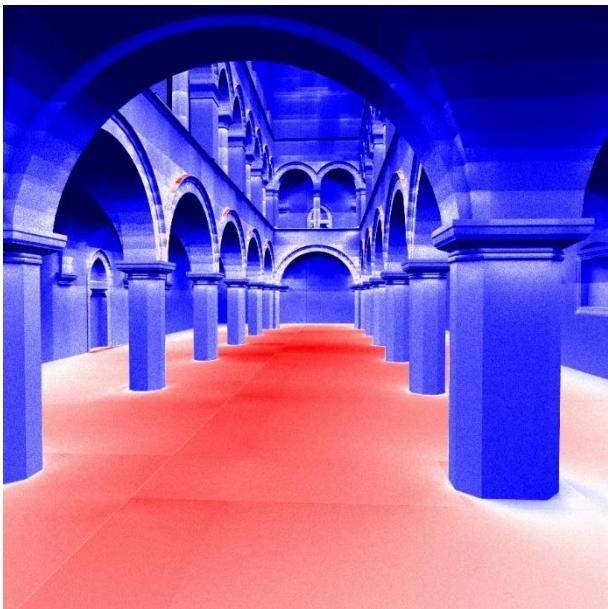
+8.2%

More

+7.6%

#TRAVESED nodes for PRIMARY rays

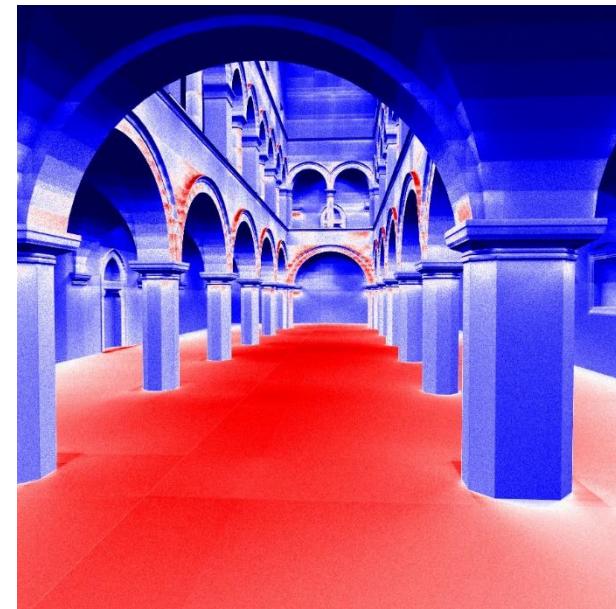
SAH



Diff(RASTERIZED, SAH)



Diff(AVERAGED, SAH)



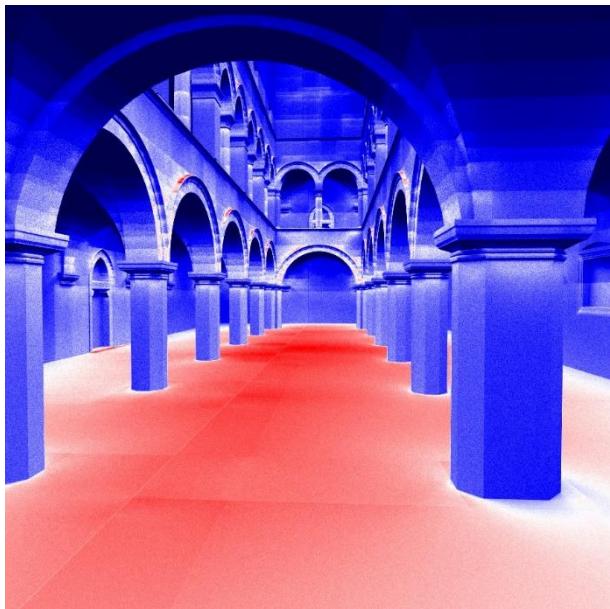
Low



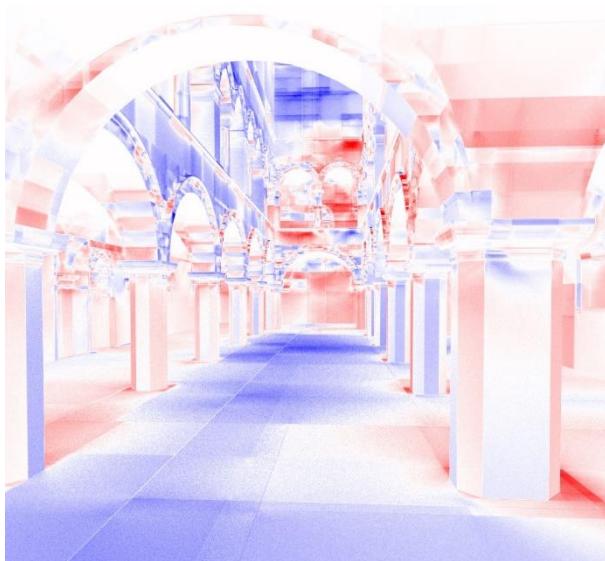
High

#TRAVESED nodes for SHADOW rays

SAH



Diff(RASTERIZED, SAH)



Diff(AVERAGED, SAH)



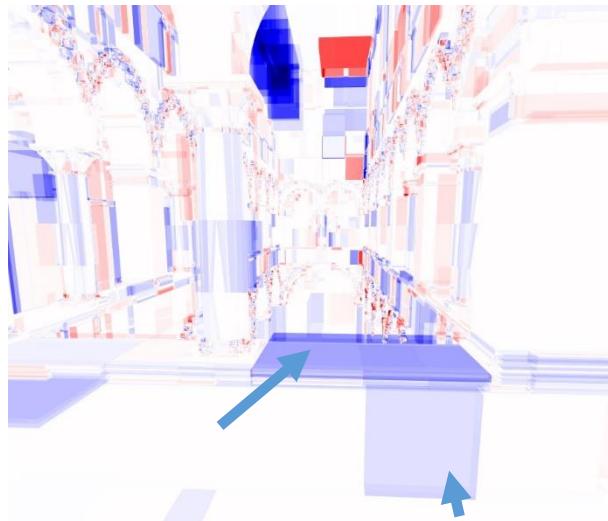
#TRAVESED nodes for SHADOW rays

SAH



281.2s

Diff(RASTERIZED, SAH)



Less

260.0s (-7.5%)

More

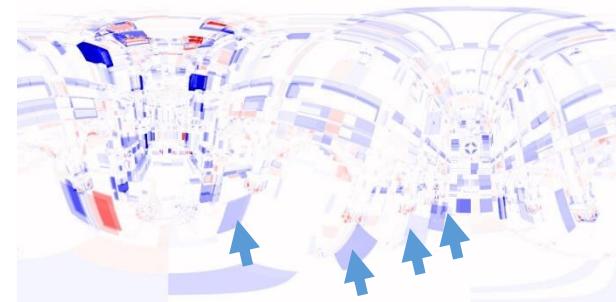
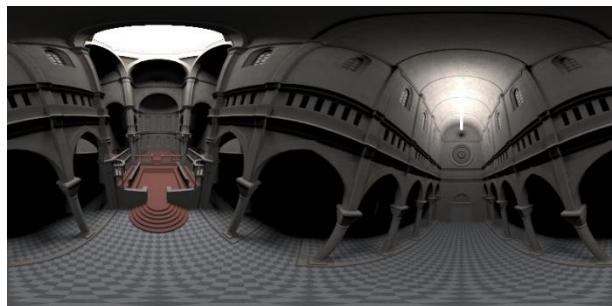
257.2s (-8.5%)

Total rendering time

SAH

Diff(RASTERIZED, SAH)

Diff(AVERAGED, SAH)



Less

More

129.9

125.8s (-3.2%)

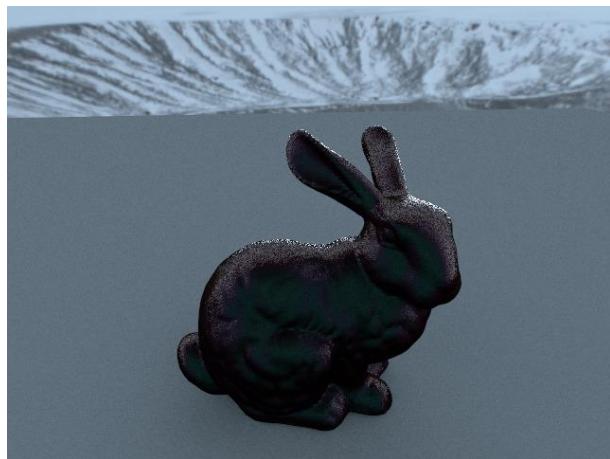
123.6 (-4.8%)

Total rendering time

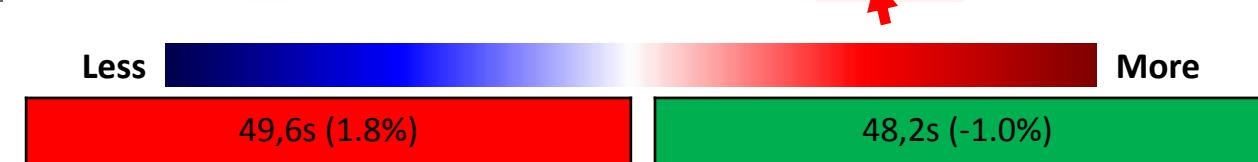
SAH

Diff(RASTERIZED, SAH)

Diff(AVERAGED, SAH)



48,7s



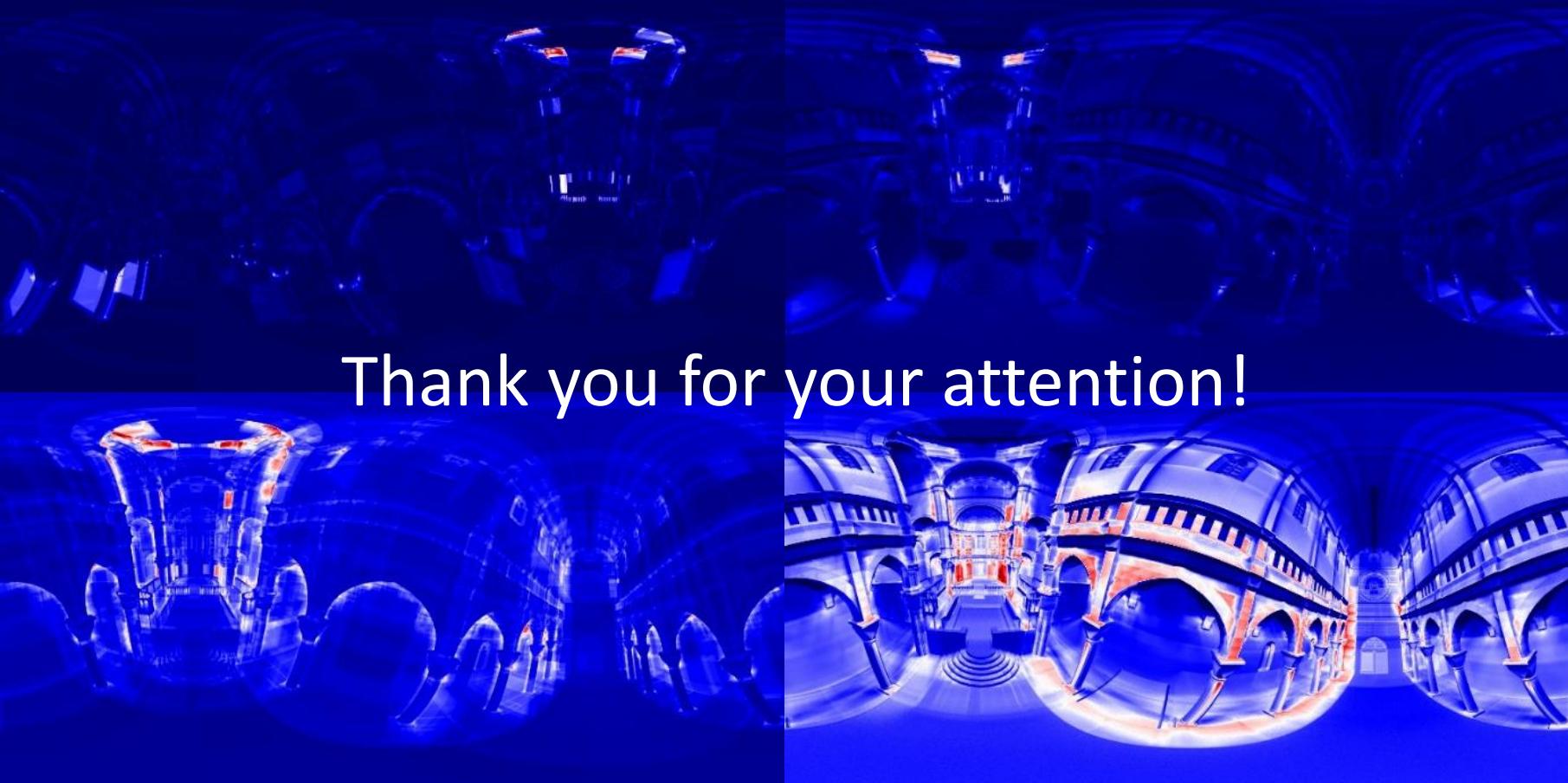
Total rendering time

Conclusions

Our RASTERIZED/AVERAGED RTSAH <> SAH

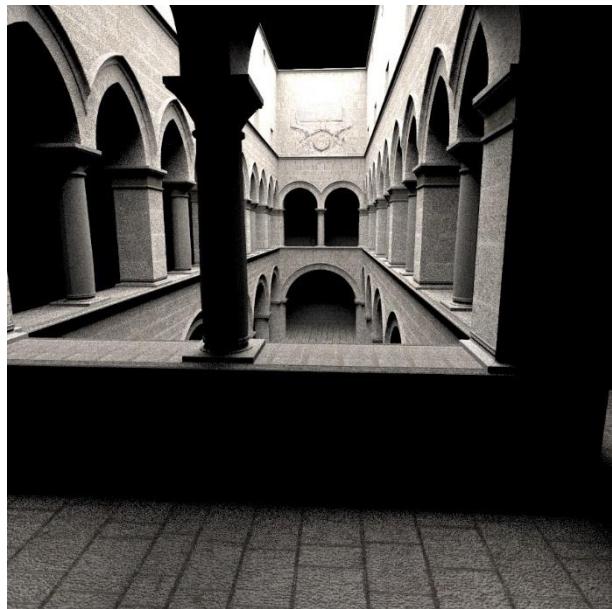
- Significant reductions in intersection tests for primary and shadow rays
- Ray termination included in build heuristic
- Same computational build complexity
- Same finite set of candidate splitting planes

Questions?



Thank you for your attention!

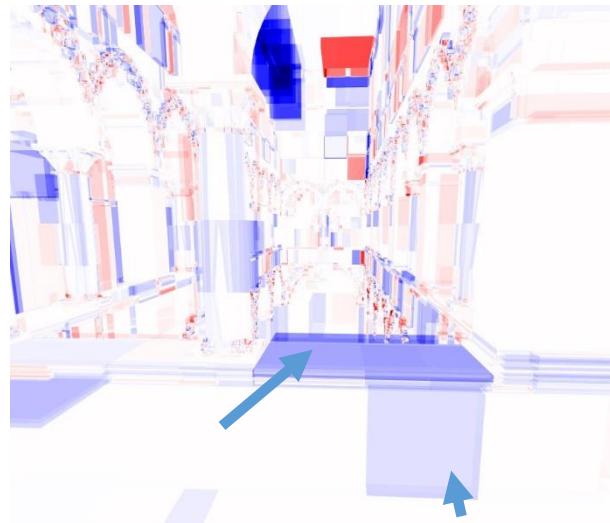
SAH



Sponza (512spp)

281.2s

Diff(RASTERIZED, SAH)



Less

-20.5%	-18.9%	+4.6%	+1.3%
--------	--------	-------	-------

260.0s (-7.5%)

More

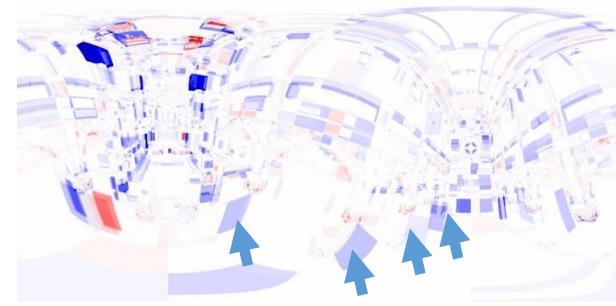
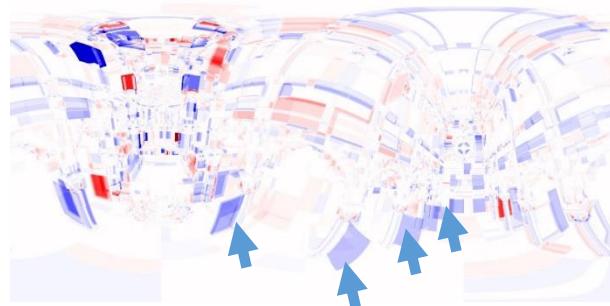
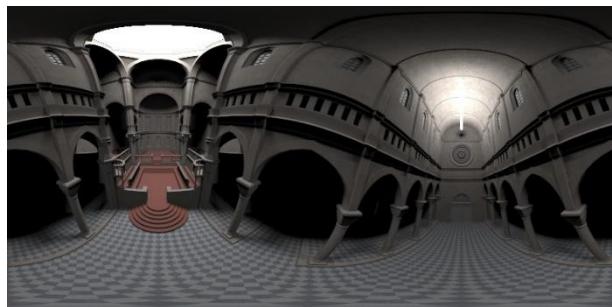
-30.7%	-40.8%	+12.2%	+10.8%
--------	--------	--------	--------

257.2s (-8.5%)

SAH

Diff(RASTERIZED, SAH)

Diff(AVERAGED, SAH)



Sibenik (512spp)

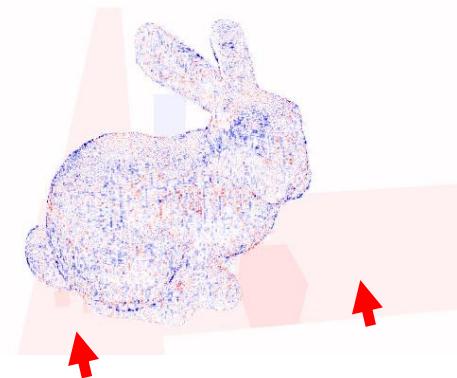
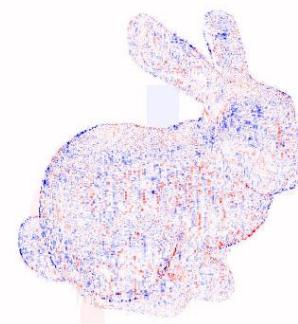
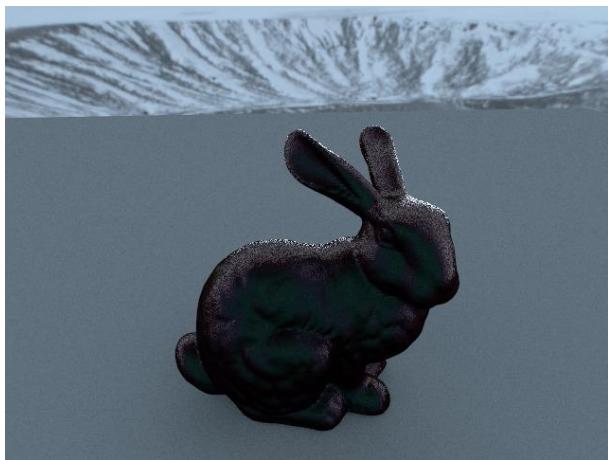
129.9



SAH

Diff(RASTERIZED, SAH)

Diff(AVERAGED, SAH)



Sibenik (512spp)

48,7s

Less

-1.8%

-1.5%

+5.5%

+7.4%

More

+0.9%

-3.7%

+2.6%

+2.7%

49,6s (1.8%)

48,2s (-1.0%)