## Supplementary material for: Combining Global and Local Virtual Lights for Detailed Glossy Illumination

Tomáš Davidovič\* Saarland University and DFKI Jaroslav Křivánek Charles University, Prague Cornell University Miloš Hašan Harvard University Philipp Slusallek Saarland University and DFKI Kavita Bala Cornell University

## 1 Stochastic Progressive Photon Mapping

In Figures 1 and 2 we provide comparison of our method with CPU implementation of Progressive Photon Mapping [Hachisuka et al. 2008] and Stochastic Progressive Photon Mapping [Hachisuka and Jensen 2009]. The timings are machine with Intel Xeon X5560 processors (with 4 physical, 8 logical cores each). Both algorithms are implemented as CPU only. To compensated for this, we the reported results are for roughly  $5 \times$  longer runtime for each method, when compared to ours. We let the algorithms run even after this and confirmed that no significant improvement could be achieved by providing slightly longer time.

## References

HACHISUKA, T., AND JENSEN, H. W. 2009. Stochastic progressive photon mapping. *ACM Trans. Graph.* 28, 5, 1–8. HACHISUKA, T., OGAKI, S., AND JENSEN, H. W. 2008. Progressive photon mapping. *ACM Trans. Graph.* 27, 5, 130:1–130:8.

Our approachProgressive Photon Mapping (PPM)Stochastic PPMImage: Stochastic Photon Mapping (PPM)Image: Stochastic Photon Photon PhotoNappi

\**e-mail:* davidovic@cs.uni-saarland.de, krivanek@ksvi.mff.cuni.cz, milos.hasan@gmail.com, slusallek@cs.uni-saarland.de, kb@cs.cornell.edu



5 min 43 sec (5k maps, 200k glob. lights, 55.6M loc. lights)

28 min 28 sec (18100M photons; 16 cores)

29 min 32 sec (8840M photons; 16 cores)

Figure 1: Results: Comparison of our method with Progressive Photon Mapping and Stochastic Progressive Photon Mapping



4 min 16 sec (10k maps, 100k glob. lights, 25.1M loc. lights)

20 min 50 sec (2175M photons; 16 cores)

21 min 9 sec (452.5M photons; 16 cores)

Figure 2: Results: Comparison of our method with Progressive Photon Mapping and Stochastic Progressive Photon Mapping