

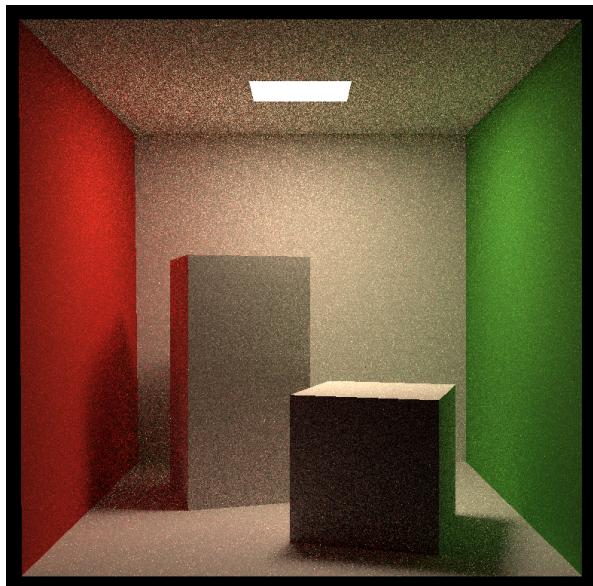
# Project 1 : Path Tracing

CG course assignment. Based on CPU calculation and C++ programming, the project uses Monte Carlo method to render the image of the Cornell Box. Integrates over all the illuminance arriving to a single point on the diffuse surfaces of objects. Repeats the integration procedure for every pixel and uses BVH structure to accelerate collision testing and multithreading to reduce time consumption. Analyzes the results under different sampling frequencies.

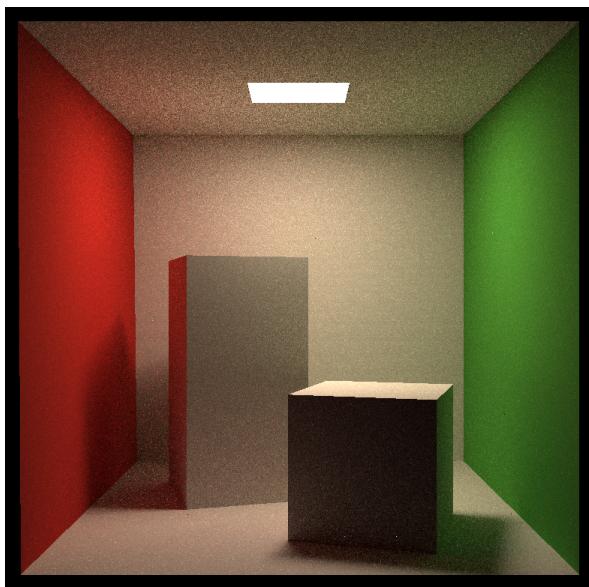
Setting : Ubuntu 18.04.2 , CPU AMD R5 3400G



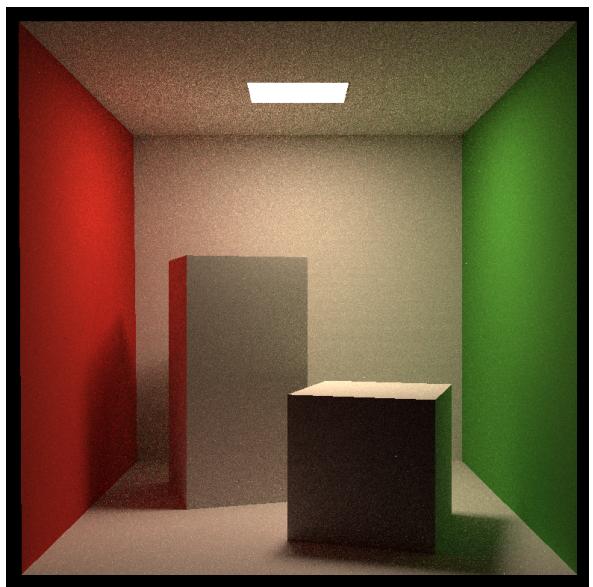
SSP16 , 4 threads , 104s



SSP16 , 1 thread , 255s



SSP64 , 4 threads , 328s



SSP64 , 1 thread , 1017s

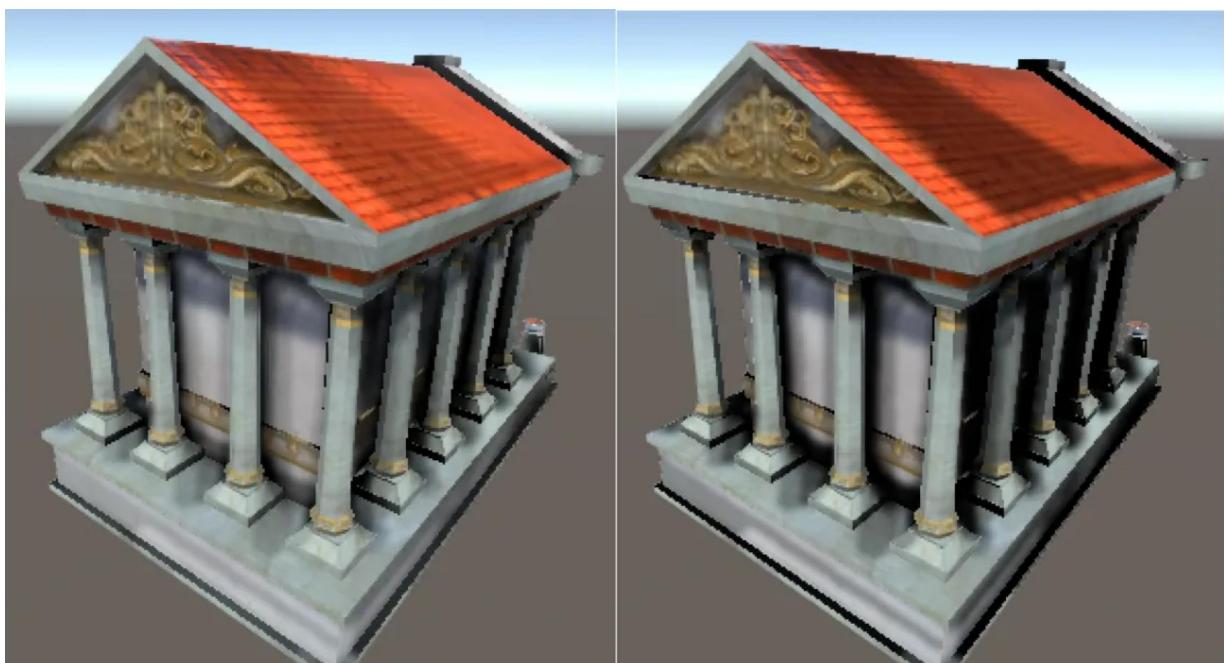
## Project 2 : Unity Shader SSAO

CG course project. Based on Unity Shader and GPU calculation, the program uses screen-space information to determine the amount of occlusion by darkening surfaces that are close to each other. Uses a scene's depth buffer in screen-space to reconstruct the coordinates of each fragment in camera space. Obtains the occlusion factor by taking multiple depth samples in a normal-oriented hemisphere sample kernel surrounding the fragment position and compare each of the samples with the current fragment's depth value. Blurs the AO texture through a bilateral filter to reduce noises and preserve edges. Analyzes the costs and compares the results.

Setting : Unity 2019.3.5f1 , GPU NVIDIA MX450 2G



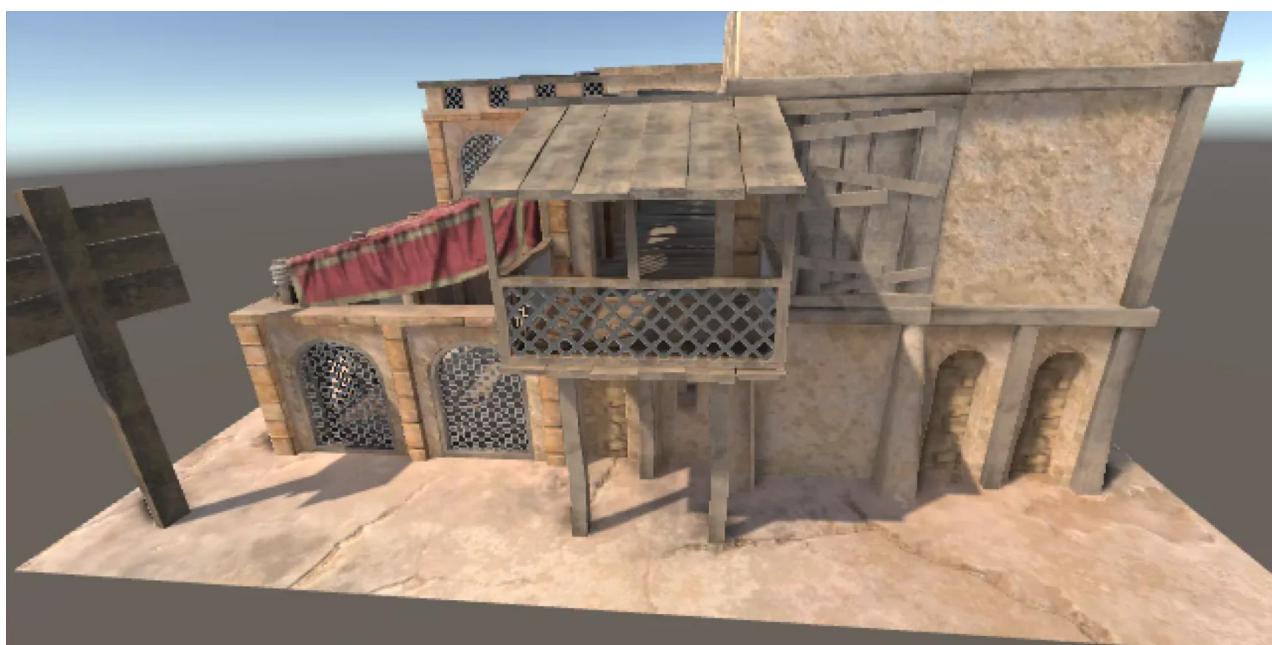
left : SSAO , right : no SSAO



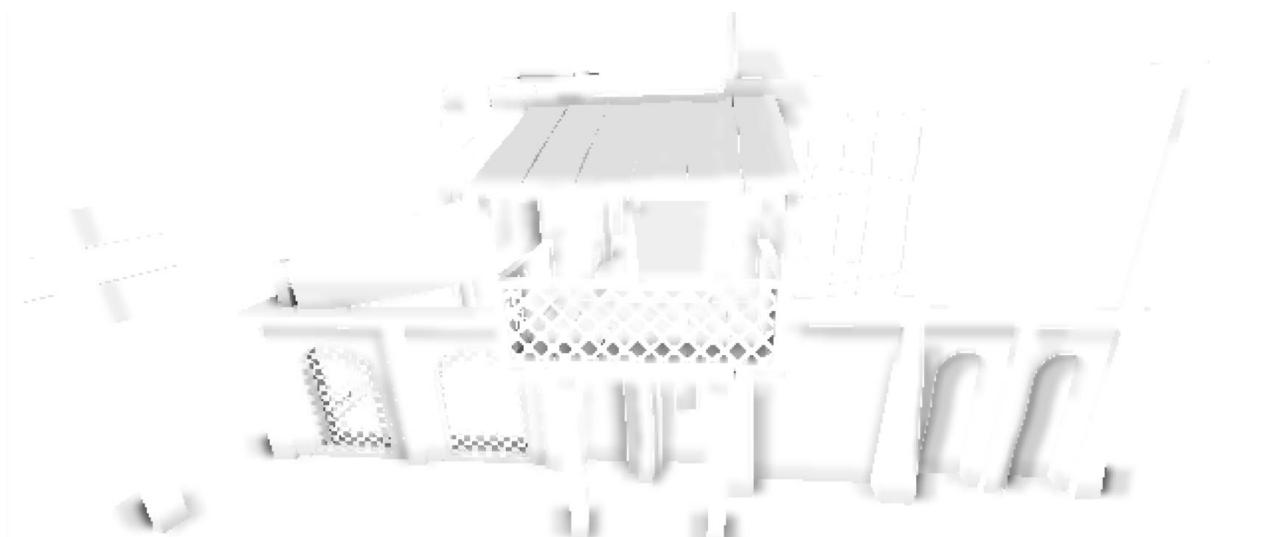
left : AO Strength=3 , right : AO Strength=5



no SSAO



SSAO



only AO