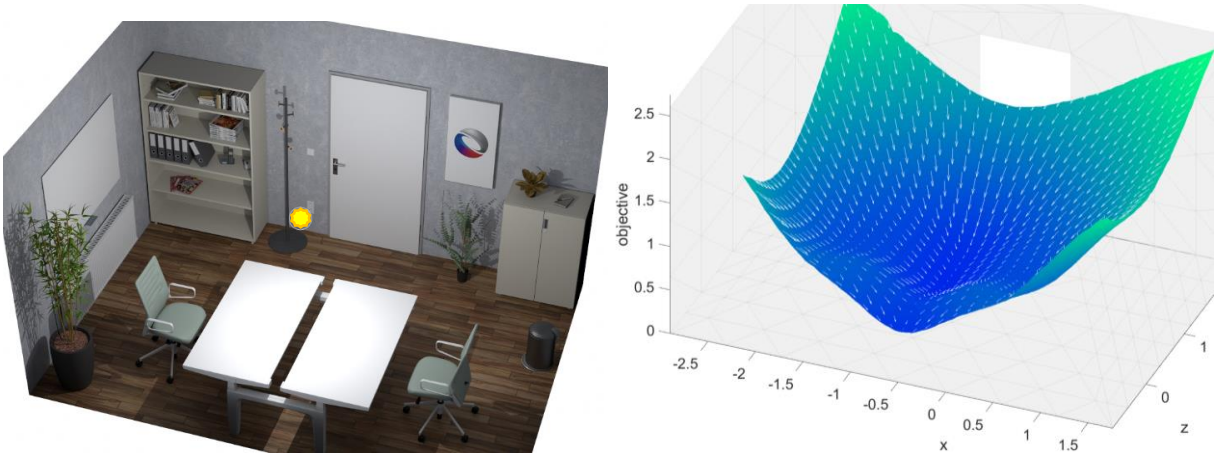
Global optimization and learning for lighting design

While modern rendering algorithms are commonly implemented in C++ and GPU APIs such as Vulkan or OpenGL, many recent advances in efficient global optimization and machine learning have produced a wide variety of available Python or Matlab toolboxes.

In this project we want to build a Python (or Matlab) interface for our rendering framework, which will enable us to run available optimization or learning methods combined with our differentiable renderer.

One application of interest is to automatically generate a lighting design – in the sense of a selection and layout of various light sources within a virtual scene – that is optimal with respect to a desired illumination target (considering, for instance, achieving sufficient and uniform illumination of office desks, energy efficiency, or artistic lighting effects). Besides automatic generation, we could also investigate a more interactive workflow, where the design system quickly updates suggested changes based on recent user inputs.

For this project, we are looking for students with good C++, Python, and/or Matlab programming skills, and interest in machine learning or global optimization methods.



Example: positioning a light source such as to illuminate two desks in a small office; right: the cost function to be optimized and its gradient. With more complex scene layouts, local minima become increasingly problematic, requiring more robust and efficient design methodology.