# Differentiable Path Tracing by Regularizing Discontinuities

Peter Quinn[1], Jérôme Parent-Lévesque[2], Cengiz Oztireli[3], Derek Nowrouzezahrai[1]

[1] McGill University, [2] Mila, Université de Montréal, [3] University of Cambridge, Google

## Abstract

Recently, viewing computer vision as an inverse rendering problem has led to a growing interest in differentiable rendering. We introduce a novel differentiable path tracing algorithm where discontinuities in the rendering process are regularized through blurring of the geometry. Our differentiable renderer implements full global illumination and has parameters for controlling the regularization, which allows for some control over the smoothness of the loss landscape. We successfully apply our system to solve several examples of challenging inverse rendering optimization problems that involve more complex light transport scenarios that cannot be handled by rasterization-based differentiable renderers.

## Background

Differentiable has found several interesting applications in recent works: optimization of scene parameters [1], inverse rendering, computer vision, and unsupervised ML such as mesh reconstruction [2]. There are two discrete, nondifferentiable steps in typical rendering that need to be handled: edges of geometry, which cause instantaneous changes in the image colour at object boundaries, and determination of the surface closest to the camera, which cause sharp changes in the image where one object overlaps another.

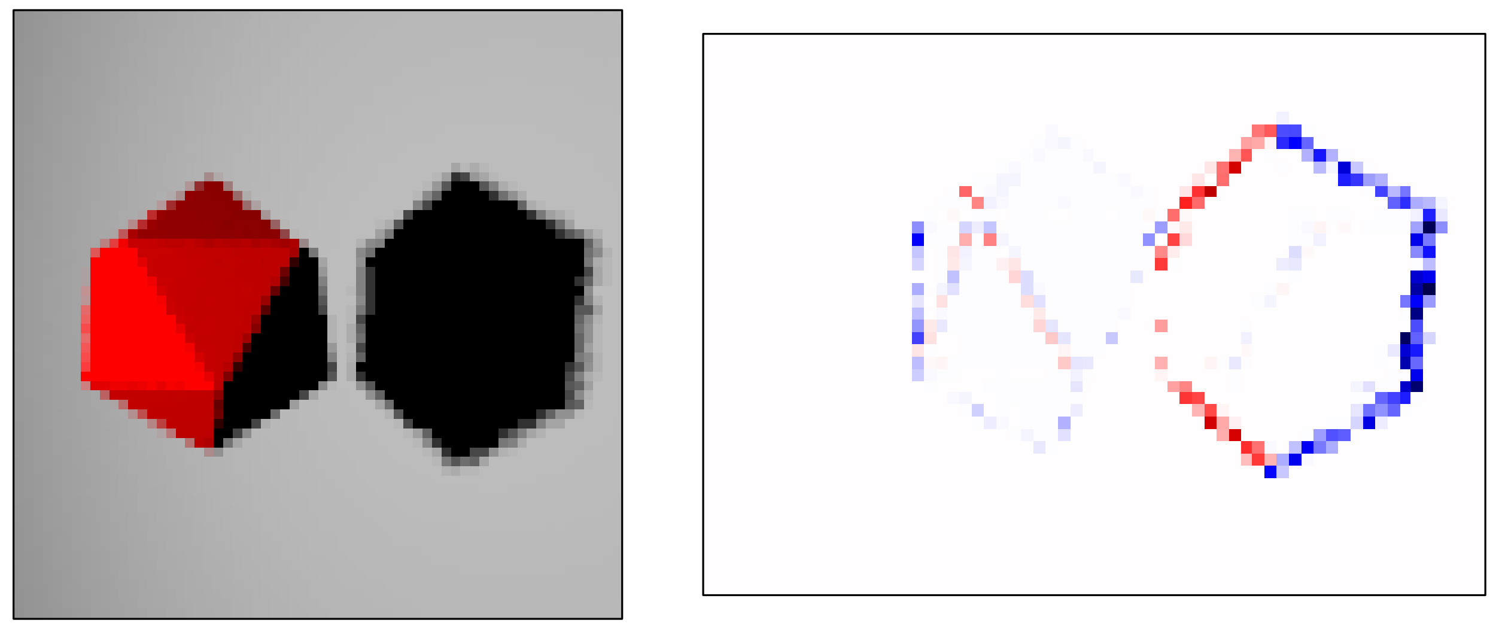
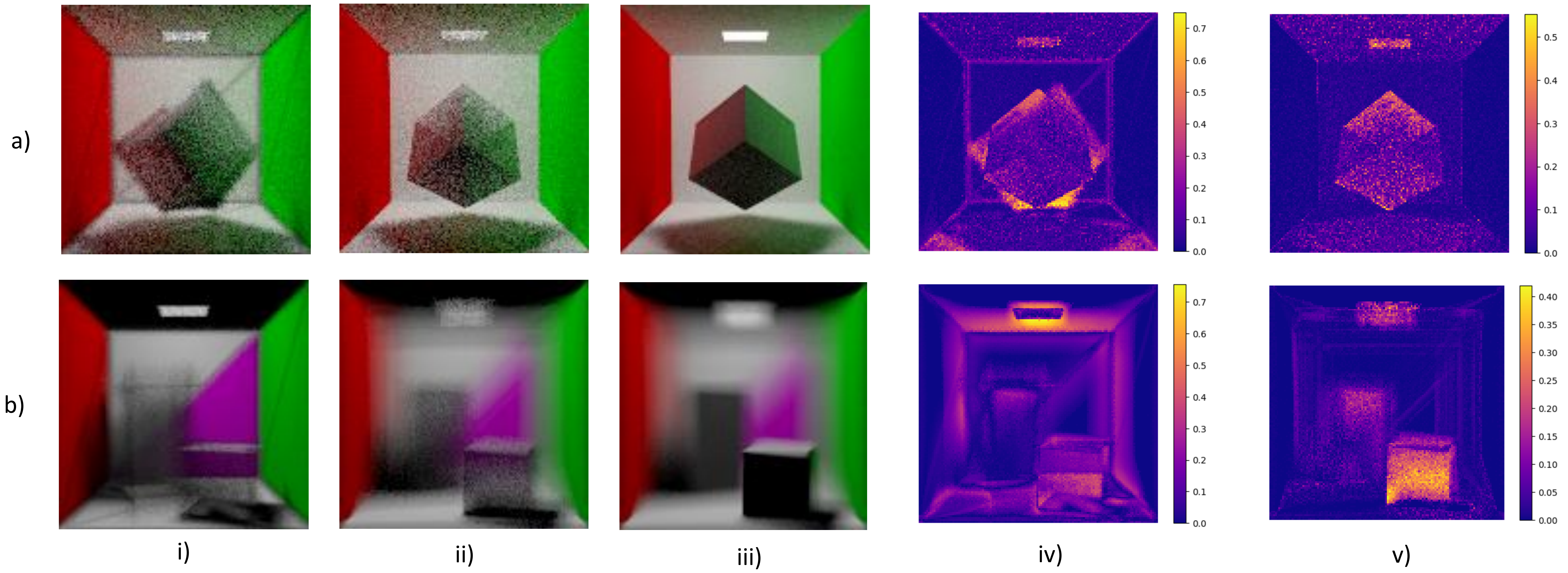


Fig. 1: An image and a rendering of the gradients for each pixel of the image given a translation of the red icosahedron to the right

## Results



**a)** A white cube in a scene with significant indirect lighting is rotated to match the target image **b)** The aperture size of the camera is adjusted to match the depth of field effect in the target **i)** Starting position **ii)** Final position **iii)** Target position **iv)** Starting error **v)** Final error

## Methodology

Edge discontinues are handled by blurring triangles according to a sigmoid function in the plane of the triangle. The depth ordering discontinuity is handled by taking a sum of all triangles along a ray weighted by their distance. We implemented a renderer in Python using the PyTorch library. PyTorch allows tracing 1000s of rays in parallel on the GPU using tensors and gradients are calculated automatically with PyTorch autograd.

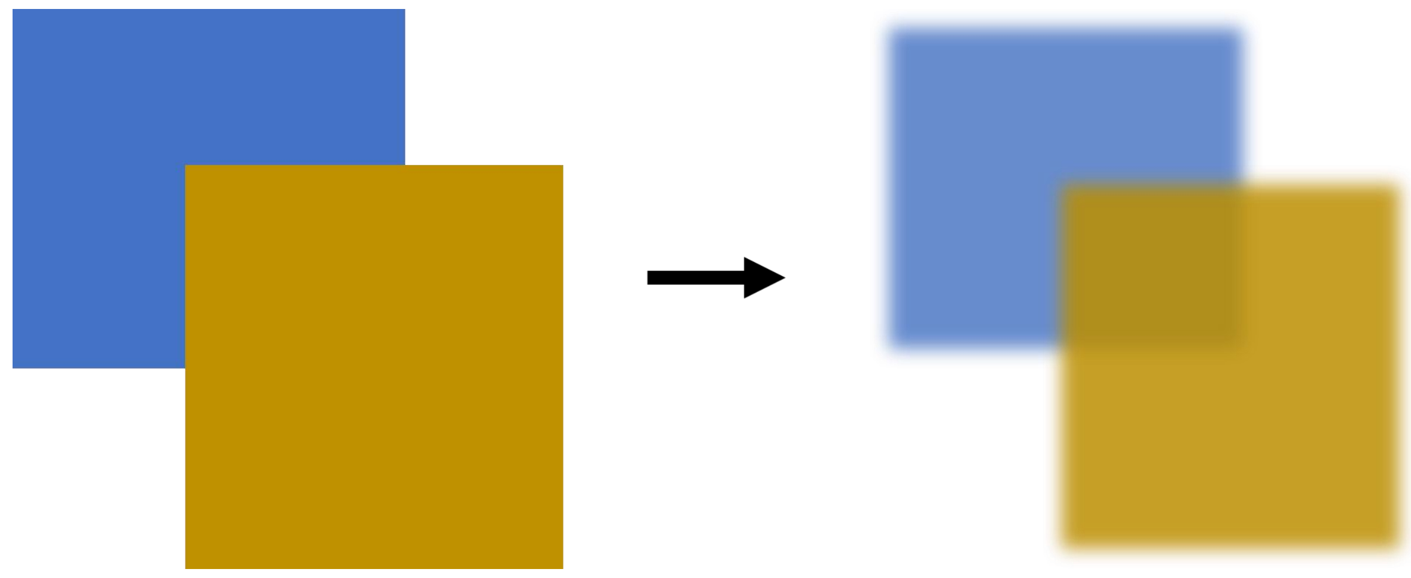


Fig. 2: **Left**: a typical image produced by a non differentiable renderer. **Right**: an example of the blur effects in the image used to make the rendering operation differentiable

## Conclusion

We present a novel differentiable approach to path tracing. We apply our approach to solve several inverse rendering problems involving higher order light transport effects, including camera effects. In future work, we would like to improve the both the efficiency of our differentiable renderer, increase the number of triangles that can be handled and add refractive materials.

**References:**

[1] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable monte carlo ray tracing through edge sampling," in SIGGRAPH Asia 2018 Technical Papers, 2018, p. 222: ACM.

[2] S. Liu, T. Li, W. Chen, and H. J. a. p. a. Li, "Soft Rasterizer: A Differentiable Renderer for Image-based 3D Reasoning," 2019.

**See more at:**

https://youtu.be/Ee678lnKDAs
https://www.peterquinn.ca/diffren/examples.html

**Acknowledgements:** McGill, NSERC, Facebook AI for PyTorch