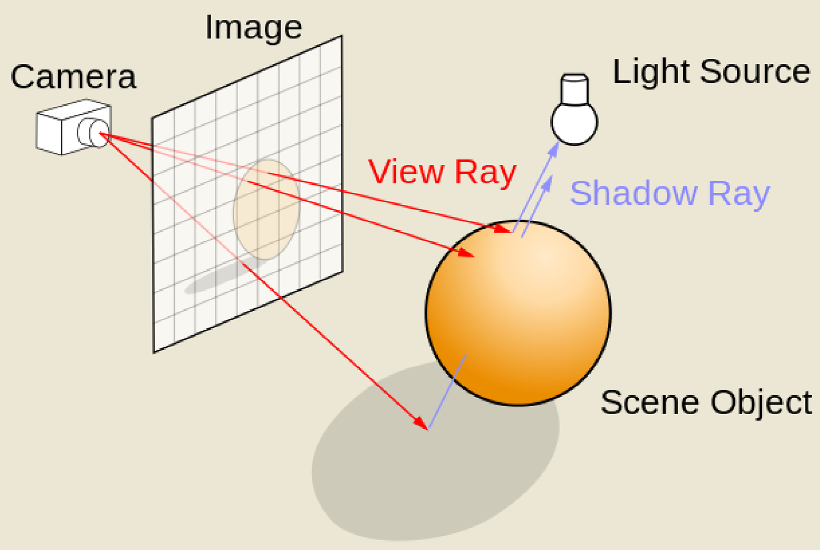
Smart Adaptive Sampling for Photorealistic Rendering: Learning Samplers for Monte Carlo Ray Tracing

Abstract

We take a machine learning based approach to adaptive sampling for Monte Carlo Ray Tracing, by using geometric and lighting data obtained through prior renders of scenes.

Motivation



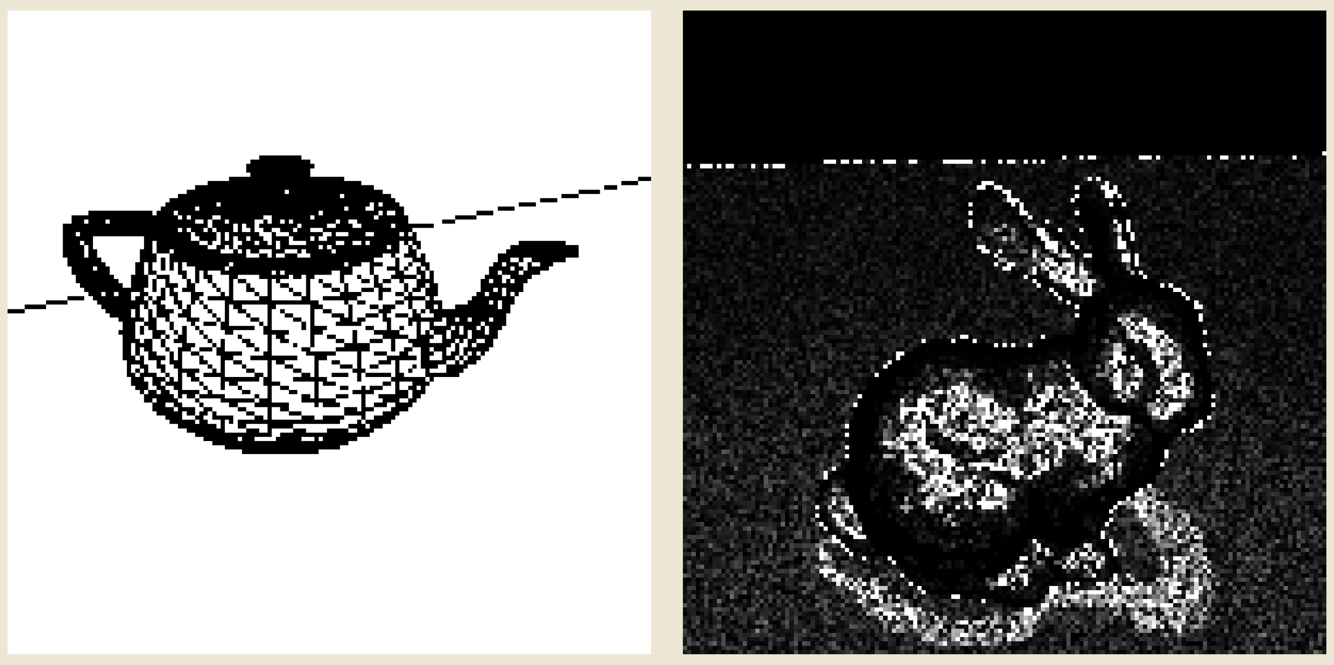
- Monte Carlo ray tracing is realistic, handles complex natural phenomena well.
- Cons: **High quality images are expensive to render.**

Adaptive Sampling

- Ideally, the number of rays for a given pixel would depend on the sampled pixel's rate of convergence to the perfect pixel.
- The challenge is thus to predict when a pixel is "close" to the perfect pixel.
- **Hypothesis: Pixel value is within convergence threshold.**

Our Approach

- Layers of Support Vector Machines to determine whether we would need to increase the number of samples.
- Implementation as **pbrt extension** (Physically Based Rendering,)
- linked with **libsvm** to solve for the SVM coefficients
  $\arg\max_\alpha \left\{ \sum \alpha_i - \sum y^{(i)} y^{(j)} \alpha_i \alpha_j K(x^{(i)}, x^{(j)}) \right\}$
- Features, labelled by **color distance to highest resolution**, normalized so that labels are balanced, include:
  - Variance in **Illuminance** of the combined ray collection
  - **Color value** of the combined ray collection
  - Differences of the 3 XYZ color channels of the two sets of ray collections
  - Difference in variance in illuminance of the two sets of ray collections
  - Number of distinct **primitives** that our combined ray collection hit



Implementation

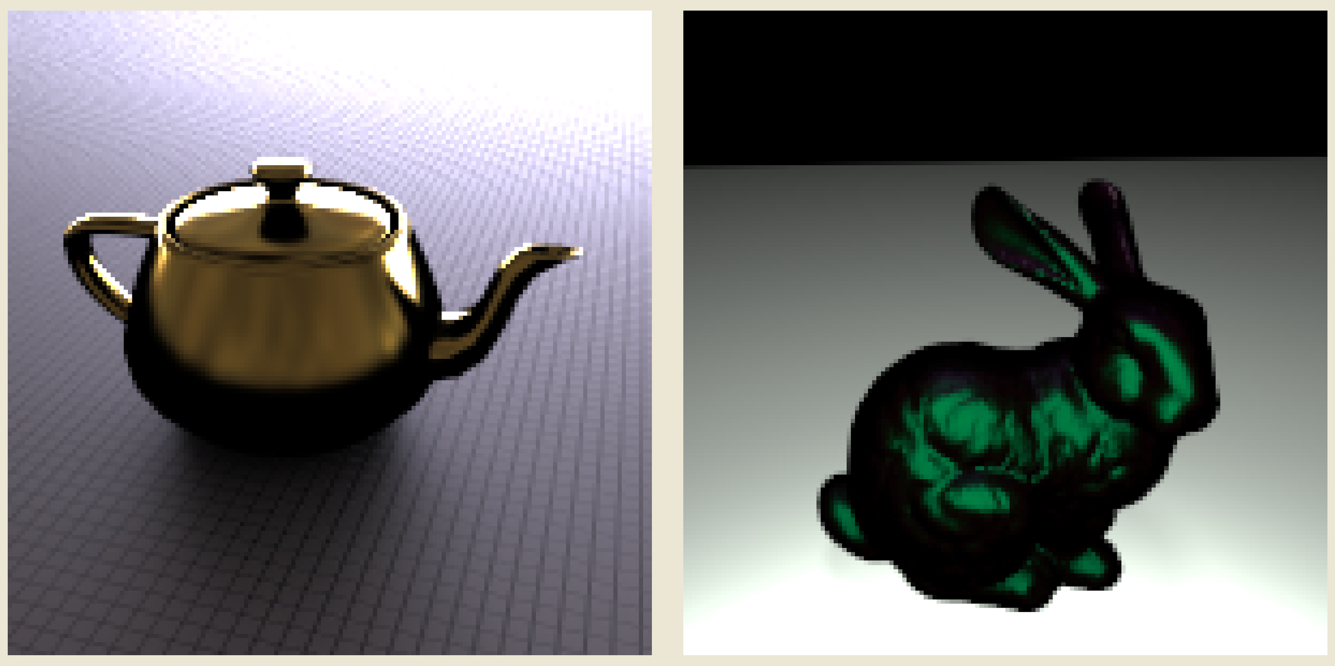
We trained our models on 4 images of 200x200 resolution. We experimented with different SVM parameters;In particular, data size and labelling thresholds were a big problem as there were a lot of support vectors.

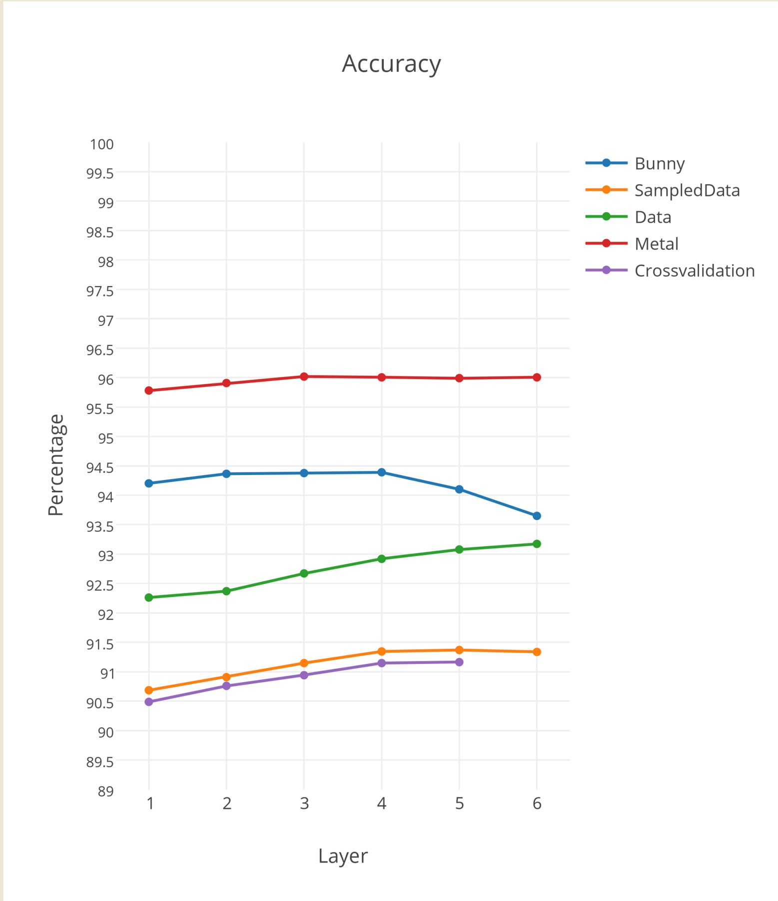
Implementation-cont.

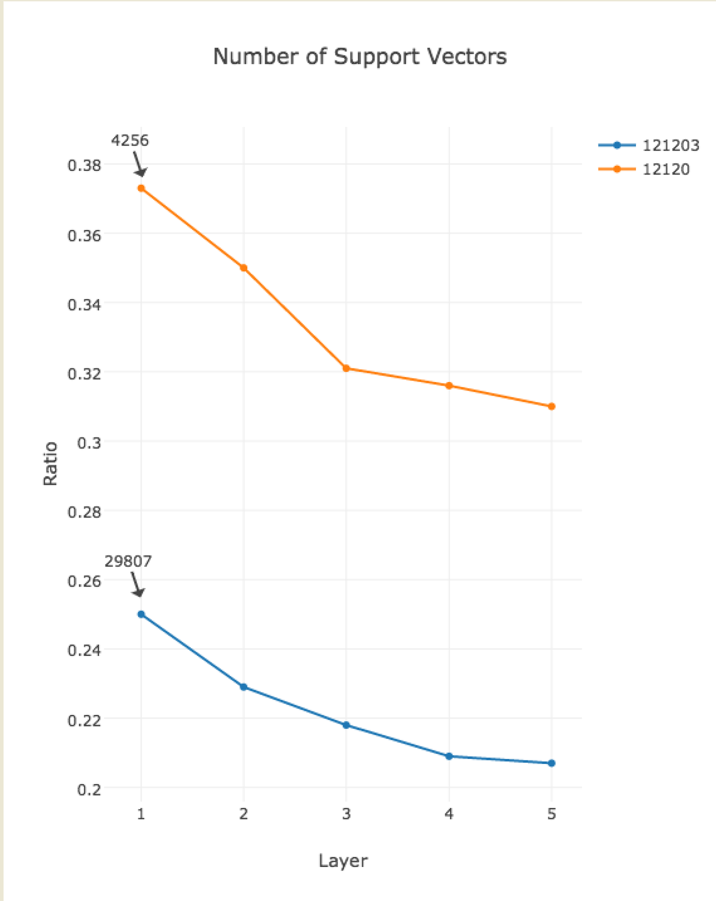
However, we obtained quite accurate results with the radial basis kernel.

Results

Here are some produced images and relevant data on our SVM models.







Future Work

- More features via better data interception
- Optimization: Ultimate goal is to make it a faster sampler
- Different labelling schemes

Acknowledgments

We would like to thank Professor Ng for the wonderful course, Albert Haque for the advice on this project, and all the TAs of 229 for making this class a great experience.

References

- Matthias Zwicker et. al.
  Recent advances in adaptive sampling and reconstruction for monte carlo rendering.
- Nima Khademi Kalantari et.al.
  A Machine Learning Approach for Filtering Monte Carlo Noise.
- Matt Pharr and Greg Humphreys.
  Physically Based Rendering: From Theory to Implementation.

Winnie Lin and Timothy Wu
CS 229 2015: Final Project
Stanford University