Space reconstruction using neural networks

There have been significant advances in machine learning over the last decade mainly driven by developments in the methods used to train neural networks.[6]. Beginning in 2006, methods to effectively train networks, sometimes using methods that had been around since the 1980s, in conjugation with better processing power and a great deal more data, made neural networks viable ways to solve problems. [5]

The goal of this project is to engineer a neural network that will generate a 3d map of a space from a photograph of a scene. This is an ambitious goal, however the first milestone involves re-implementing a recent paper titled "3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction" by Choy et al. from the Computational Vision and Geometry Lab at Stanford. Given this project as starting point in addition to the freely available source code on github, there will always be results to analyze. The only thing differentiating possible analyses is the quality of results. The original network is implemented using theano, a deep learning framework used mostly in research [1]. I plan to re-implement the network in using a more production oriented framework tensorflow [7]. The point of this is that it will allow me to gain a through understanding of how the network works, which in turn will allow me to modify it to suit my needs.

Tensorflow has low level features that allow for a great deal of control over the architecture of a network. For example, it is possible to specify networks at a greater resolution compared to something higher level like keras[3], which can serve as a wrapper for both theano and tensorflow. It is possible to specify the nature of the individual units that make up a network, even allowing for the construction of custom units.

Research like this is motivated by the recent rise in the number of complicated problem machine learning has been able to tackle in a lot of different fields. An obvious application for this project would be better spatial reasoning and navigation for robots and self-driving vehicle.

Currently the plan is to use Amazon Web services (AWS) for training the neural network.

AWS has a few gpu enabled machine images (AMIs), the cheapest among them being the "p2.xlarge". It costs about 1 dollar per hour, but the rate varies with in a certain range. I will be using the dataset used in the original paper,shapenet[2], an online repository of 3d models. The dataset requires about 30 GBs of space. The effectiveness of the re-implementation compared to the original model on the shapenet dataset will serve as a metric for judging the quality of the re-implemenation. Further more it will inform subsequent modifications.

References

[1] James Bergstra et al. "Theano: A CPU and GPU math compiler in Python". In: *Proc. 9th Python in Science Conf.* 2010, pp. 1–7.

[2] Angel X Chang et al. "Shapenet: An information-rich 3d model repository". In: *arXiv preprint arXiv:1512.03012* (2015).

[3] François Chollet et al. *Keras*. `https://github.com/fchollet/keras`. 2015.

[4] Christopher B Choy et al. "3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2016.

[5] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press, 2016.

[6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *Nature* 521.7553 (2015), pp. 436–444.

[7] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.