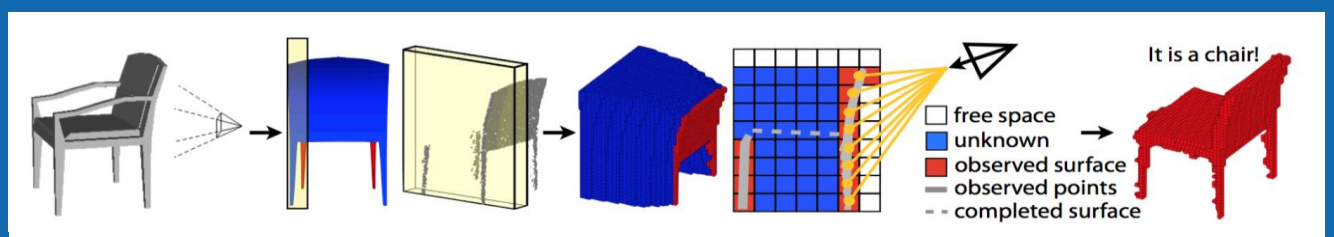
3D Object Recognition with Deep Networks

Students: Adrian Schneuwly, Johannes Oswald, Tobias Grundmann

Supervisors: ***Martin Oswald, Pablo Speciale***

1 Goal of our work: Reimplement Voxnet [1]



2 Method Overview

Input Data / Preparation

- ModelNet10/40 dataset - 3D CAD models of 10/40 common object categories with 100 unique models each (.mat files)

Fig. 2: voxelized models; Source: Voxnet paper [1]

- A 3D shape is represented as 32 x 32 x 32 voxel grid (Fig. 2).
- *Contribution*: Converted multiple .mat files to a single highly compressed hdf5 file, which contains the complete dataset.

Deep Convolutional Neural Networks

Task: Object Recognition as a Classification Problem (Fig. 3)

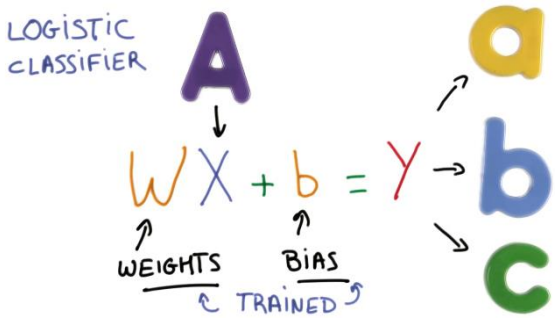


Fig. 3: Classification problem; Source: Udacity [4]

- Neural Network: Non-linear activation function applied to input to create non-linear output (Fig. 4)

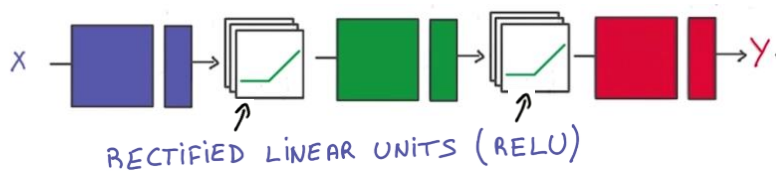


Fig. 4: Relu; Source: Udacity [4]

- Deep Neural Network: Multiple Connected Layers of weights, which are trained
- Convolutional Nets: Convoluting multiple voxel of one layer into a stack of voxel or a activation map (Fig. 5a)

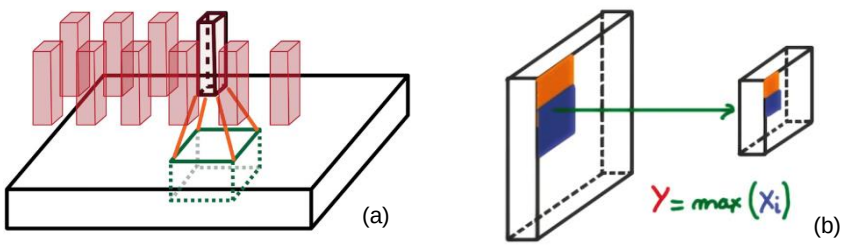


Fig. 5: (a) Convolution, (b) Max-Pooling; Source: Udacity [4]

- Max-Pooling: Non-linear down sampling by choosing maximum values of rectangles created from segmenting the volume (Fig. 5b)

3 Voxnet & Implementation

Sequence of multiple Convolutional layers, Max Pooling layers followed by Fully Connected layer

- CNN ~ 900k parameters
- Activation: Leaky ReLu

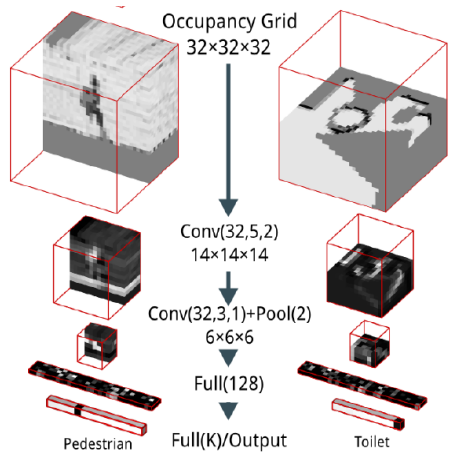


Fig. 6: VoxNet layers; Source: VoxNet paper [1]

Contribution:

- The Convolutional Neural Network was re-implemented in Python using the Keras framework with Theano backend.

4 Results & Conclusion

Training

- The training process takes around 9 to 20 hours on a NVIDIA GTX 980TI (6GB) GPU depending on the size of the dataset

Results

- Our implementation (ETH VoxNet) achieves similar result as the original VoxNet[1] (Table 1).
- Classification accuracy coincide with the original authors approach for ModelNet10, but for Modelnet40 a significant overfitting was observed.
- A possible explanation for the bad performance could be that the data was not augmented for multiresolution, since the training time was limited.

Algorithm	ModelNet10	Modelnet40
VoxNet [1]	83%	92%
3DShapeNets [2]	77%	83.5%
ETH VoxNet	**81.8%**	**82.4%**

Table 1: Classifation accuracy

5 References

[1] D. Maturana and S. Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. *International Conference on Intelligent Robots and Systems (IROS2015)*, 2015.

[2] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang and J.Xiao. 3D ShapeNets: A Deep Representation for Volumetric Shape; Proceedings of 28th IEEE Conference on Computer Vision and Pattern Recognition

[3] http://sun.cs.princeton.edu/

[4] Udacity Deep Learning course