3DCNN-DQN-RNN: A Deep Reinforcement Learning Framework for Semantic Parsing of Large-scale 3D Point Clouds

Fangyu Liu[1,*] **, Shuaipeng Li** [1,*] **, Liqiang Zhang** [1,⋆]**, Chenghu Zhou** [2,⋆] **,**
Rongtian Ye[1] **, Yuebin Wang**[1] **, Jiwen Lu** [3]
∗ indicates equal contribution; ⋆ indicates corresponding authors.

[1]*Faculty of Geographical Science, Beijing Normal University, Beijing, China*
[2]*Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences*
[3]*Department of Automation, Tsinghua University, Beijing, China*

ICCV17
International Conference on Computer Vision 2017

Overview

LeCunn, Bengio, Hinton: ***"We expect the future progress in vision to come from systems that are trained end-to-end and combine ConvNets with RNNs using reinforcement learning to decide where to look."***

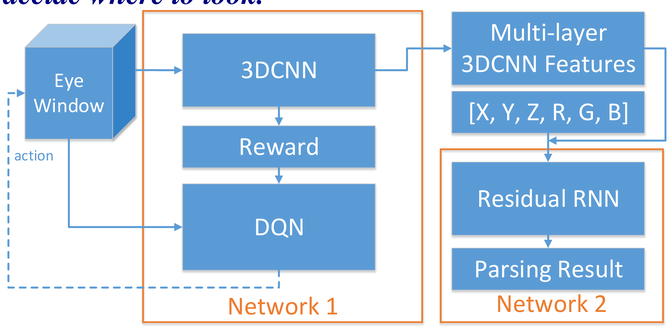

Figure 1: A sketch of our proposal.

We fuse the 3D convolutional neural network (CNN), Deep Q-Network (DQN) and Residual recurrent neural network (RNN) for an efficient semantic parsing of large-scale 3D point clouds. An **Eye Window** under control of the 3D CNN and DQN can localize and segment the points of the object's class efficiently.
(**Eye Window** is an intelligent agent (a 3D bounding box) that changes size and position to capture certain target.)

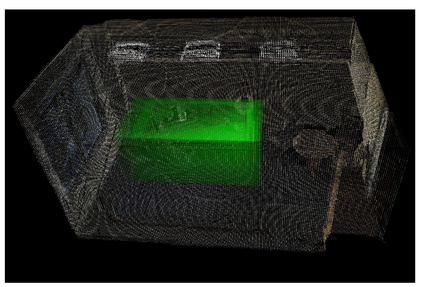

Figure 2: A snapshot of **Eye Window** searching tables.
Scan the above QR code to watch a demo video.

Network Details

Through supervised learning, the 3D CNN learns features about shape, spatial relationship, color and context of a certain kind of point cloud object, and then encode them into a discriminative feature representation called 3D CNN feature. Besides, the 3D CNN outputs reward vectors as DQN's input:

$$f(E(p,q)) = \theta^{\top} E(p,q) = 2R_1 + 1 - R_2 = r \tag{1}$$

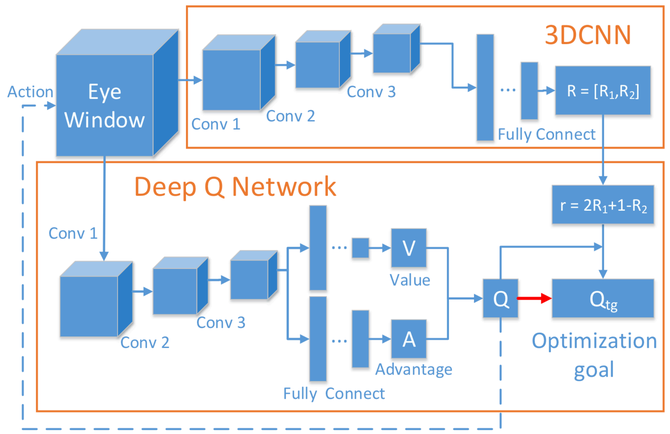

Figure 3: Part 1: 3DCNN+DQN

To parse certain object from a large scene, an **Eye Window** traverses the whole data for localizing the points belonging to a certain class.The process can be expressed as the following optimization problem:

$$E(p_b, q_b)^* = argmax(f(E(p,q))) \tag{2}$$

Our goal is to estimate a position p_b and size q_b of Eye Window in the scene to maximize $f(E(p,q))$.
The size and position of Eye Window are controlled by the DQN, which is a reinforcement learning network that acts as the "brain" of our network. The training/learning process is identical with the original DQN brought up by DeepMind. Following the tradition, we define Q, Q_{tg} for DQN:

$$Q(s,a;\theta,\alpha,\beta) = V(s;\theta,\beta) + A(s,a';\theta,\alpha) - \frac{\sum_{a'} A(s,a;\theta,\alpha)}{\|A\|} \tag{3}$$

$$Q_{tg} = \tanh(\sum_{t=0}^{N-1} \lambda^t r_t + \lambda^N Q') \tag{4}$$

The network parameters get updated when the current Q does a gradient descent to Q_{tg}:

$$\theta_{T+1} = \theta_T + \lambda(Q_{tg} - Q(s,a;\theta_T))\nabla_{\theta_T} Q(s,a;\theta) \tag{5}$$

During the traversal, the DQN gets the probability that Eye Window contains its target (computed through the reward vector output by the 3D CNN). If the feedback is unsatisfactory, it traverses all adjacent regions and determines a region that is worth looking at. It then directs Eye Window to change size and move towards that region.
We re-apply the 3D CNN to compute the reward vector of the points in the new Eye Window... Repeat the process until Eye Window accurately envelops the target.

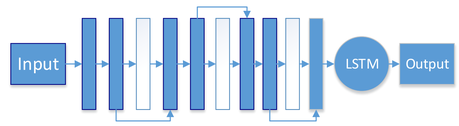

Figure 4: Part 2: RNN

Once the target is localized and enveloped, the 3D CNN feature, coordinate and color of each point within Eye Window, are combined into a vector, which is input into the Residual recurrent neural network (RNN). The RNN further abstracts and merges feature representations of the points in Eye Window. The output of RNN is the classification result.

Eye Window keeps moving until all points of all classes of objects are parsed.

Result

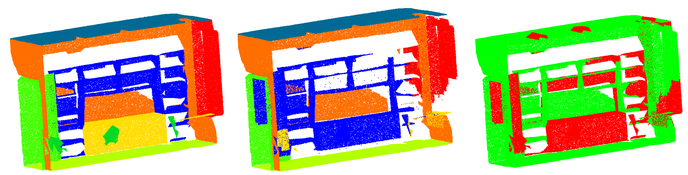

Figure 5: Parsing result of a room. Green part in third pic indicates predicted right and red wrong.

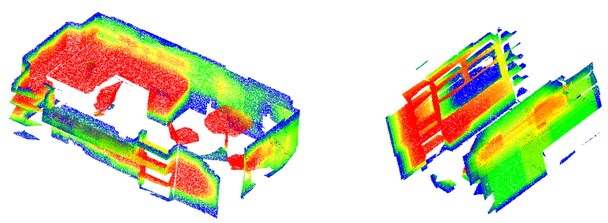

Figure 6: Activity thermodynamic diagram of an eye window.