Particle Picking

For particle picking, we apply reference-free Difference of Gaussian (DoG) image transform. We obtain the DoG map via subtraction of two versions of Gaussian filtered images and the peaks detected in the DoG are selected to be potential particles. Each of them has the size of 28*28*28 voxels. We identified DoG parameter settings that maximize precision and recall for detecting particles in the first 9 tomograms. Then, we directly apply the DoG model to the testing data and select the potential complexes and their coordinates. In summary, we select out 4114 potential subtomograms in the testing data. Some of them are assumed to be the imaging background which contains nothing. Note that we did the same thing in the 9 training tomograms and select 13 types of subtomograms for downstream classification task. (One class contains nothing; the other 12 classes refer to the 12 given PDB identificators.)

Subtomogram Classification

After we obtain the potential training subtomograms using 9 annotated tomograms. We train a supervised Convolutional Neural Network (CNN) for classification. We concatenate eight 3D 3x3x3 convolutional layers and each layer is activated by ReLU. Five max pooling layers are mixed among the convolutional layers. At end, two fully connected layers with 50% dropout are added and a softmax activation is appended.

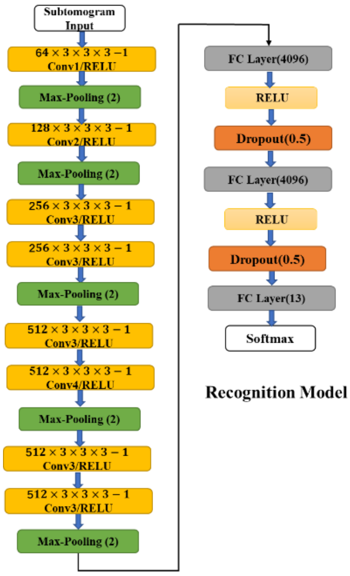

Figure 1: Subtomogram recognition model: '64x3x3x3-1 Conv' represents a 3D convolutional layer with 64 3x3x3 filters and stride of 1. 'ReLU' and 'Softmax' are activation layers. 'Max-Pooling (2)' means that max operation is implemented over 2x2x2 regions with stride of 2'. 'FC Layer 1024' represents fully connected layers with 1024.

The model is trained using stochastic gradient descent optimizer. We try to minimize the categorical cross-entropy cost function by adding Nesterov momentum of 0.9. In addition, the initial learning rate is set at 0.005 with a decay factor of 1e-7. The training processes are performed with a batch size of 64 for 100 epochs.

Then, we apply the learned model directly to the testing data and obtain their labels. We omit the samples which are claimed to be the imaging background and only list the coordinates and labels for samples from the other 12 classes.