Computational Biology

Project 1: Automated Particle Picking in Cryo-EM

An Ju **Yangkun Zhang** **Tianxiao Shen**

Institute for Interdisciplinary Information Sciences

April, 2016

1 Introduction

This project is to pick particles from micrograph in cryo-electron microscopy (cryo-EM) [3]. Given a micrograph, our system will predict the center coordinates of particles in it.

We address this problem in a two-step procedure: first, we scan the whole micrograph by a sliding window and judge whether it contains a particle or not [2]; and then, for all the windows containing a particle, we compute their center and merge the ones nearby to get the final results.

An Ju programmed the major part. Yangkun Zhang assisted in coding and did program optimization, including efficient GPU implementation. Tianxiao Shen was in charge of algorithm design, report and demo.

2 Algorithm

2.1 Window Classification

To judge whether a window contains a particle or not is a binary classification problem. We use a convolutional neural network (CNN) to deal with this task, which is one of the most powerful machine learning algorithms for image classification [1] .

2.1.1 Preprocessing

padding

We preprocess all the values to make them range between 0 and 1, by subtracting min and then dividing by $\max - \min$. We directly train our network on these values.

2.1.2 Network Architecture

Our network contains 5 learned layers—3 convolutional and 2 fully-connected. The convolutional layers have ? kernels of size $\times$, ? kernels of size $\times$, and ? kernels of size $\times$

respectively. Each of them is followed by a max-pooling layer. The first fully-connected layer has ? neurons, and the second has 2 neurons, corresponding to the two classes.

We use Rectified Linear Units (ReLUs) nonlinearity, and the final layer is fed to a 2-way softmax to produce a Bernoulli distribution. We use the cross-entropy between the true and predicted distribution as our loss function. And we use stochastic gradient descent (?) with each micrograph as a batch to train our model.

2.1.3 Training Data

As the two classes are skewed—most windows do not contain a particle, we adopt a simple strategy to get balanced training data: all golden particles are used as positive cases (each particle determines a window by locating its center), and we randomly sample negative cases of the same amount.

2.2 Merge Neighboring Windows

A particle could be contained by multiple overlapping windows, and thus we need to merge them into a single one. For a window predicted to have a particle in it, we compute the coordinates of its center and find its nearest neighbor among all the particles picked so far. If the distance between them is less than a threshold d, we merge them in the center of mass way; otherwise we regard it as a new particle.

We calculate the confidence of a particle by summing up the confidence of its components. If this value is greater than a threshold C, we report it as a final predicted particle.

The overall procedure is described in Algorithm 1.

3 Experiments

3.1 Parameters Setting

3.2 Results

Acknowledgments

We thank Zhipeng Jia for the helpful discussion with him.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Algorithm 1 Particle Picking

Input: a micrograph g

Output: a list l of particles

$l, \hat{l} \leftarrow \emptyset$
for all window w in g **do**
  $(p_0, p_1) \leftarrow \text{CNN}(w)$
  if $p_1 > p_0$ **then**
    $(x_w, y_w) \leftarrow \text{center}(w)$
    $(x, y, m, c) \leftarrow \text{findNearestNeighbor}((x_w, y_w), \hat{l})$
    if $\text{dist}((x_w, y_w), (x, y)) < d$ **then**
      $(x, y) \leftarrow (\frac{mx+x_w}{m+1}, \frac{my+y_w}{m+1})$
      $(m, c) \leftarrow (m + 1, c + p_1)$
    else
      $\hat{l} \leftarrow \hat{l} \cup \{(x_w, y_w, 1, p_1)\}$
    end if
  end if
end for
for all $(x, y, m, c) \in \hat{l}$ **do**
  if $c > C$ **then**
    $l \leftarrow l \cup \{(x, y)\}$
  end if
end for

[2] R. Langlois, J. Pallesen, J. T. Ash, D. N. Ho, J. L. Rubinstein, and J. Frank. Automated particle picking for low-contrast macromolecules in cryo-electron microscopy. *Journal of structural biology*, 186(1):1–7, 2014.

[3] M. Liao, E. Cao, D. Julius, and Y. Cheng. Structure of the trpv1 ion channel determined by electron cryo-microscopy. *Nature*, 504(7478):107–112, 2013.