Vertex reconstruction with Graph Neural Network in JSNS^2

Changhyun Yoo[1,*] and *Junghwan* Goh[1,**] on behalf of the JSNS^{2}collaboration

[1]Department of Physics, Kyung Hee University, Seoul 02447, KOREA

Abstract. The JSNS^2(J-PARC Sterile Neutrino Search at the J-PARC Spallation Neutron Source) experiment searches for neutrino oscillations at 24m baseline with the J-PARC's 3 GeV 1 MW proton beam incident on a mercury target at the Materials and Life science experimental Facility (MLF). The JSNS^{2}detector consists of three cylindrical layers, an innermost neutrino target, an intermediate gamma-catcher, and an outermost veto. The neutrino target is made of 17 tonnes of Gd-loaded LS (Gd-LS) stored in an acrylic vessel, 3.2m(D) $\times$ 2.5m(H). The detector consists of a total of 120 photomultiplier tubes (PMTs), 96 PMTs for inner and 24 PMTs for veto. In JSNS^2, a maximum likelihood method based on the PMT charges is used to reconstruct position and energy of the event. We introduce Static Graph Convolution Neural Network (SGCNN), which is a combined model of PointNet and Graph Neural Network (GNN). The model was trained by Monte Carlo (MC) samples, and the position and charge of 96 inner PMTs was used as the training feature.

1 Introduction

The primary objective of the J-PARC Sterile Neutrino Search at J-PARC Spallation Neutron Source (JSNS^2) experiment is searching for sterile neutrinos at $\Delta m^2 \sim 1\text{eV}^2$, a direct test of the LSND anomaly, by measuring neutrino oscillations at a baseline of $\sim$ 24m.The detector is placed at the 3rd floor of the Materials and Life science experimental Facility (MLF) building at J-PARC as shown in the Figure 1[1].

The JSNS^2 detector is cylindrical shaped neutrino detector. It consists of three layers : the target in the innermost, the gamma-catcher in the intermediate, and the veto layer in the outermost region. The target is an acrylic vessel of its dimension as 3.2m in diameter and 2.5m in height, filled with 17 tonnes of gadolinium-loaded liquid scintillator (Gd-LS). The gamma-catcher and veto layers are filled with 31 tonnes of pure liquid scintillator (LS), housed in a stainless steel vessel of 4.6m in diameter and 3.2m in height. To collect lights produced in the target region, 96 photomultiplier tubes (PMTs) are installed in the gamma-catcher region. Additional 24 PMTs are installed in the veto region for the cosmic ray background rejection[2].

Energy and position at the interaction vertex are reconstructed with the charged collected in each PMTs. In JSNS^2, event reconstruction is done based on the maximum likelihood method. In several neutrino experiments such as IceCube[3], JUNO[4] and DUNE[5], event reconstruction algorithms using deep learning method is widely adopted. In this work, we developed a deep learning model for vertex position reconstruction of JSNS^2 MC simulation.

*e-mail: yewzzang@khu.ac.kr

**e-mail: jhgoh@khu.ac.kr

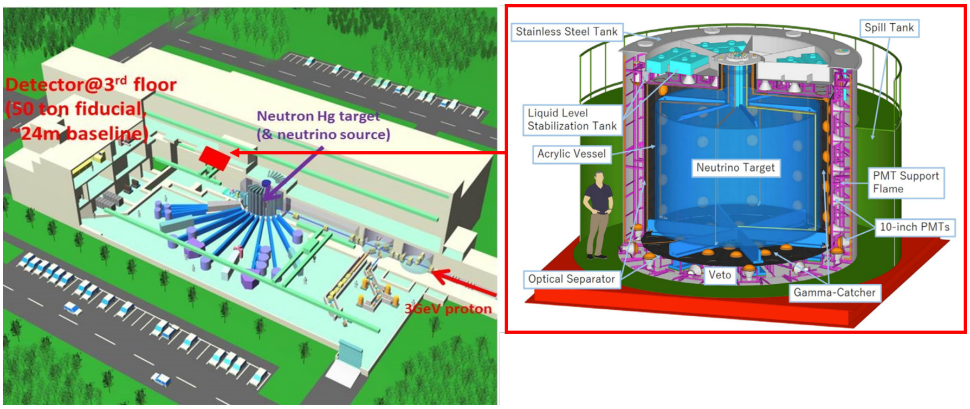

Figure 1. A bird eye view of MLF and JSNS^{2}detector. The detector is located in third floor of MLF

2 Deep Learning Model Architecture

In general, vertex position can be reconstructed because more charges are expected in a PMT when the vertex is closer to it, and *vice versa*. Therefore one can develop algorithms to reconstruct the position of the vertex (x, y, z) using the position of i-th PMT in the detector (x_i, y_i, z_i) and the collected charge (Q_i), addressing complicated detector geometry and optical characteristics of each elements.

We constructed our deep learning model, modifying the Dynamic Graph Convolution Neural Network (DGCNN)[6]. The EdgeConv layer of the DGCNN captures local geometric structure from general relationships of elements by embedding them as graphs of points and edges, and performing convolution-like operations on the features of surrounding points. In DGCNN, EdgeConv layer aggregates information from graphs built dynamically using the k-nearest neighbors clustering algorithm (k-NN) in the feature-space.

The output of EdgeConv layer $\tilde{F}_i$ for i-th PMT is calculated as Equation 1,

$$\tilde{F}_i = \sum_{j=1}^{k} \mathrm{MLP}(F_i, F_i - F_j) \tag{1}$$

where F_i denotes feature variables of i-th PMT. The input feature of i-th PMT is (Q_i, x_i, y_i, z_i). MLP stands for the multi-layer perceptron. The summation runs over the items from the k-NN clustering. Unlike the original DGCNN, we used only the PMT coordinates in real space in the clustering. Therefore, our model can be considered as Static Graph CNN (SGCNN). After three EdgeConv layer calculations, all features are combined through a MLP, max-pooling, and a MLP layer to extract the desired feature variables, (x, y, z).

3 Model Training with Monte Carlo Simulation

To train our model, we rely on the Monte Carlo (MC) simulation samples. We used Reactor Analysis Tool (RAT)[7], a simulation based on GEANT4[8] and GLG4Sim[9] libraries. JSNS^2's detector geometry and PMT information were used in our simulation. Positron with fixed energy, ranging from 1 to 10 MeV, were generated uniformly inside the acrylic tank target region. A total of 101 000 samples were generated, with 10 100 samples for each energy.

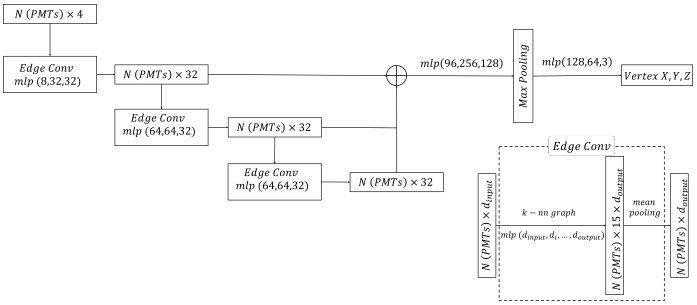

Figure 2. Diagram of our deep learning model architecture.

After removing events generated outside of the active region, a total of 100 798 MC samples were used for model training.

The samples were split into three sets: training, validation, and evaluation, with a ratio of 6:2:2. We used the loss function to be the LogCosh[10] function, in regression of vertex position. The ADAM [11] optimizer was used with the learning rate parameter to be 10^{-4}. The training is done for 3000 epochs, with the batch size of 256.

Figure 3 is a training and validation loss curve. We chose the model parameters at the minimum validation loss at 1978th epoch and conducted evaluations of the model performance.

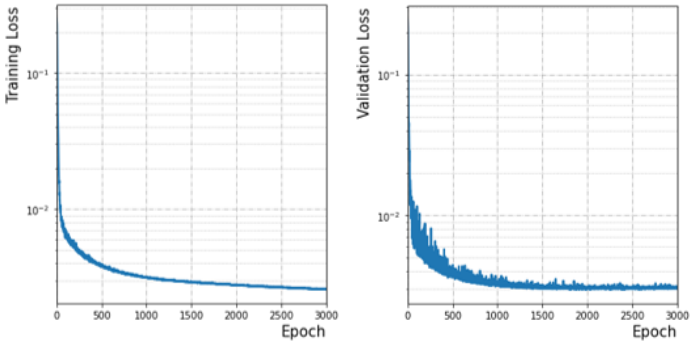

Figure 3. Loss curves (Left : training set. Right : validation set)

4 Results

Figure 4 shows the distribution of reconstructed vertex in X-Y and R^2-Z planes. The red lines indicate the border of the acrylic tank. Most of the reconstructed vertices are located inside of the acrylic tank. Figure 5 shows the difference between reconstructed vertex and true vertex. Gaussian fitting was performed on the results of evaluation samples covering the entire energy range (1 to 10 MeV) together. Detailed numbers of mean(μ) and sigma(σ) value of Gaussian fitting can be found in the Table 1. Figure 6 shows the Vertex Resolution as a function of energy.

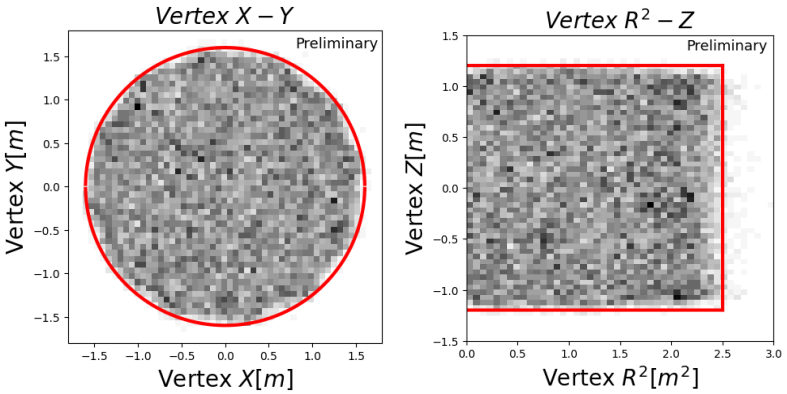

Figure 4. Reconstructed Vertex Distribution. (Left : X-Y Plane Right : R^2-Z plane)

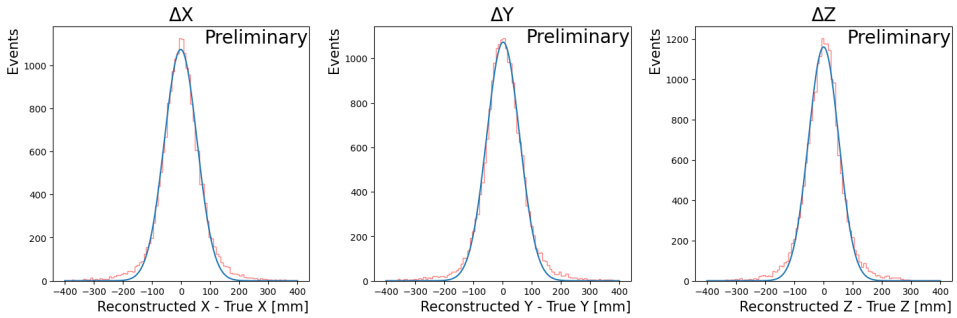

Figure 5. Reconstructed Vertex Resolution. (Left : ΔX, Center : ΔY, Right : ΔZ)

	mean(μ)	sigma(σ)
ΔX	-1.157 ± 0.021	56.24 ± 0.021
ΔY	1.871 ± 0.021	56.43 ± 0.021
ΔZ	-0.031 ± 0.019	51.96 ± 0.019

Table 1. Gaussian fit results of vertex resolution.

5 Conclusion

In this work, we introduced our deep learning model with convolution on static graphs, SGCNN, for vertex reconstruction was carried out by making a positron MC sample. Our SGCNN model shows a good performance in vertex reconstruction using only charge and PMT position. In addition, we expect improvements if more events and feature information such as pulse timing is used in the further studies.

Acknowledgement

We acknowledge the support by the National Research Foundation of Korea (NRF) Grants No. 2022R1A5A1030700, 2020R1C1C1008082, 2020K1A3A7A0908013112.

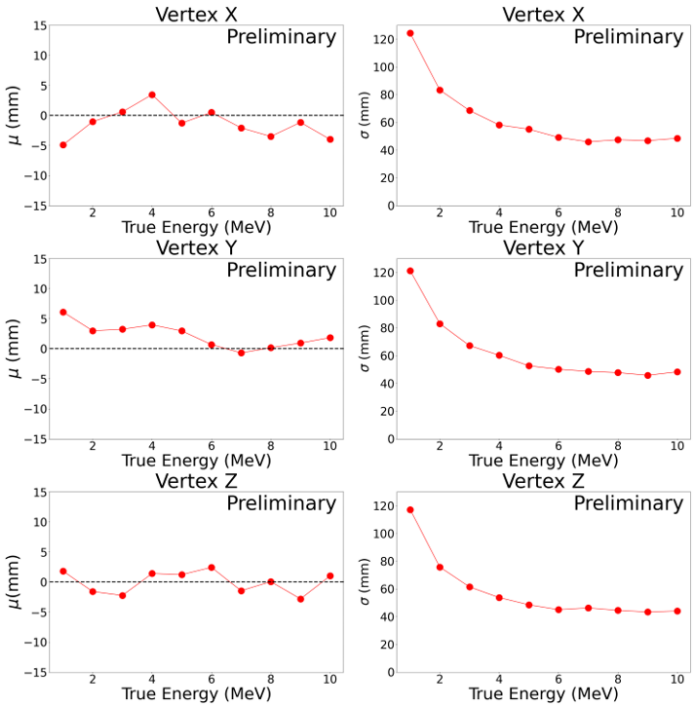

Figure 6. Reconstructed vertex resolution as a function of energy. (Left column : the mean(μ) value of Gaussian fitting, Right column : the sigma(σ) value of Gaussian fitting)

References

[1] Ajimura et al., arXiv preprint arXiv:1705.08629 (2017)

[2] Ajimura et al., Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **1014**, 165742 (2021)

[3] R. Abbasi et al., Journal of Instrumentation **17**, P11003 (2022)

[4] Qian et al., Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **1010**, 165527 (2021)

[5] I. Seong, L. Hertel, J. Collado, L. Li, N. Nayak, J. Bian, P. Baldi, *Convolutional neural networks for energy and vertex reconstruction in DUNE*, in *33rd Conference on Neural Information Processing Systems (NeurIPS), Machine Learning and the Physical Sciences Workshop* (2019)

[6] Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein, J.M. Solomon, ACM Transactions on Graphics (tog) **38**, 1 (2019)

[7] S. Seibert et al., *Rat-pac* (2014), `http://rat.readthedocs.io/en/latest/`

[8] S. Agostinelli et al. (GEANT4), Nucl. Instrum. Meth. A **506**, 250 (2003)

[9] G. Horton-Smith et al., *Introduction to glg4sim* (2006), `https://www.phys.ksu.edu/personal/gahs/GLG4sim/`

[10] S. Jadon, *A survey of loss functions for semantic segmentation*, in *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)* (IEEE, 2020), pp. 1–7

[11] D.P. Kingma, J. Ba, arXiv preprint arXiv:1412.6980 (2014)