Kinematic Analysis Using Multilayer Perceptron for 6-DOF Manipulator Control System

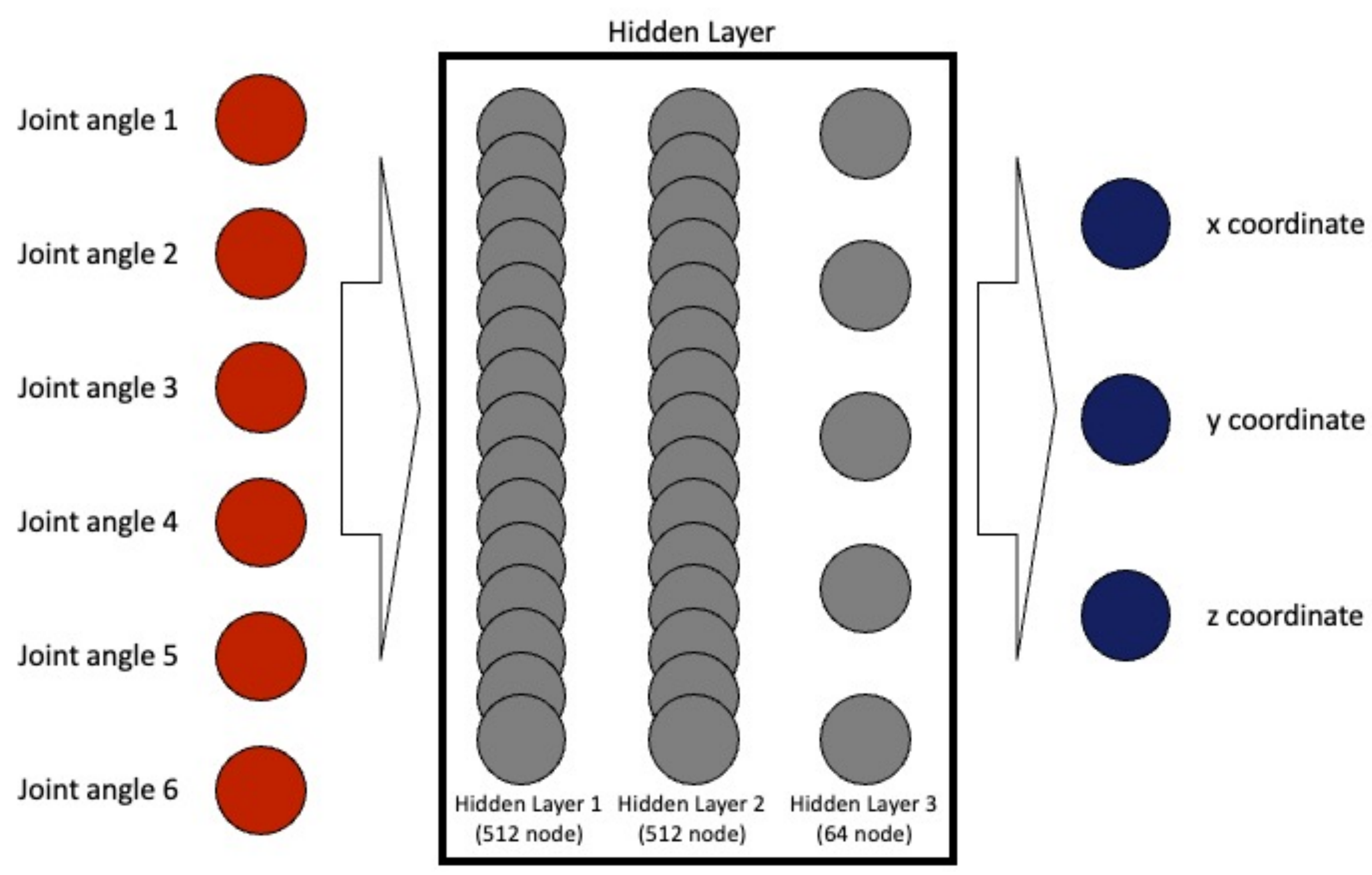
Wonmook Jeong*
1600, Chungjeol-ro, Byeongcheon-myeon, Dongnam-gu, Cheonan-si, Chungcheongnam-do, Republic of Korea
**Corresponding author: jwonmook@gmail.com*

I Introduction

- The use of manipulators is increasing in fields such as remote work and factory automation, and a kinematic analysis of the model is required for precise control of the manipulator.
- Using deep learning to solve problems in the analysis field of mechanical engineering and promote follow-up research.
- In this study, the kinematic model of the 6dof manipulator is analyzed.

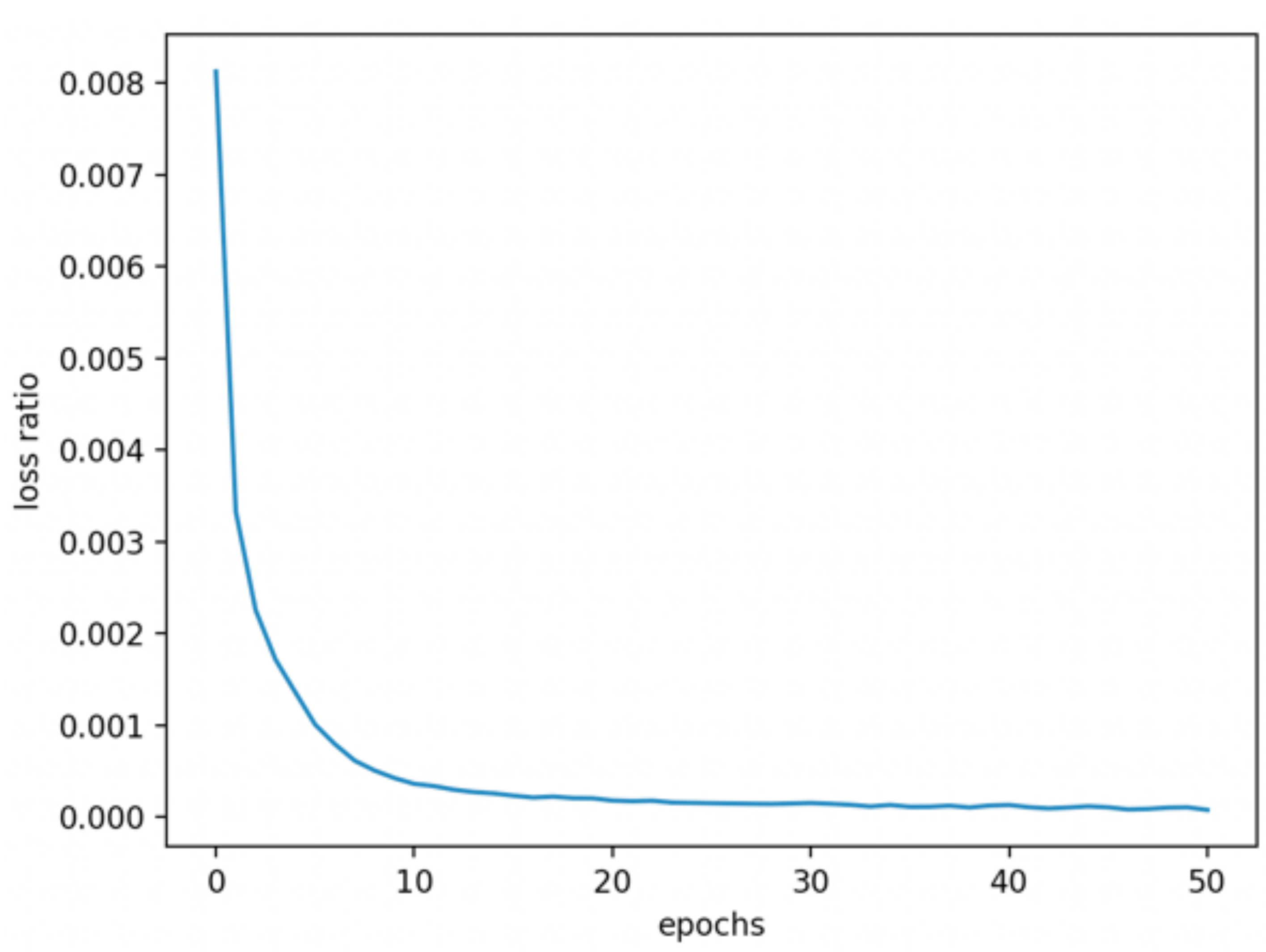
II MLP (Multi Layer Perceptron)

- By analyzing the forward kinematics of the manipulator, the position coordinates (x, y, z) of the end effector must be obtained through 6 joint angles.
- The input layer consists of 6 nodes with joint angles, and the output layer consists of 3 nodes with position coordinates x, y, and z. The hidden layer consists of three layers, each of which consists of 512, 512, and 64 nodes.



Clustering and Project bounding box

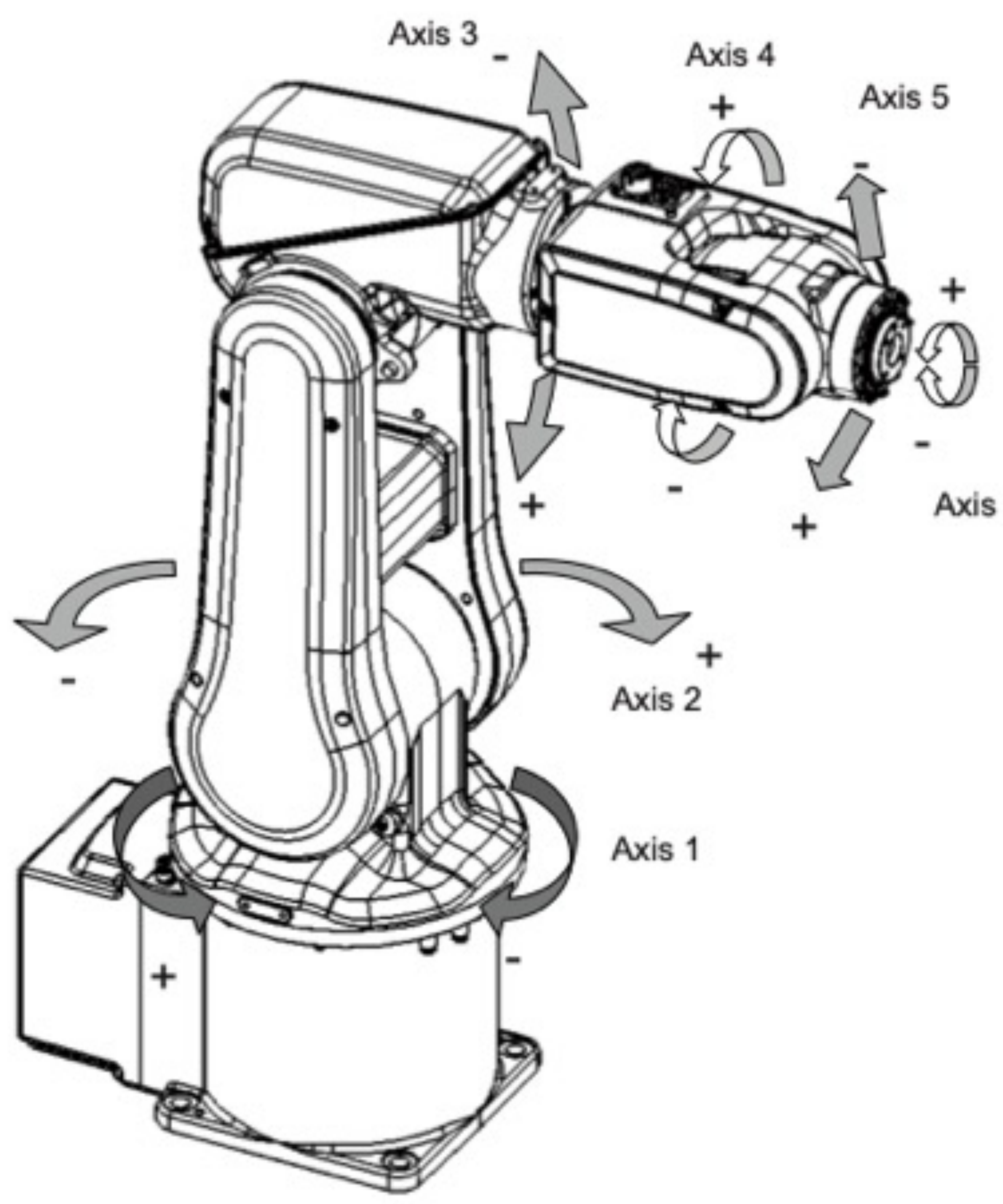
- The optimizer trained using Adam with a learning rate of 0.001 and trained 50 epochs.
- The loss function used MSE(Mean square error) provided by Pytorch.



Loss ratio graph

III Training Data

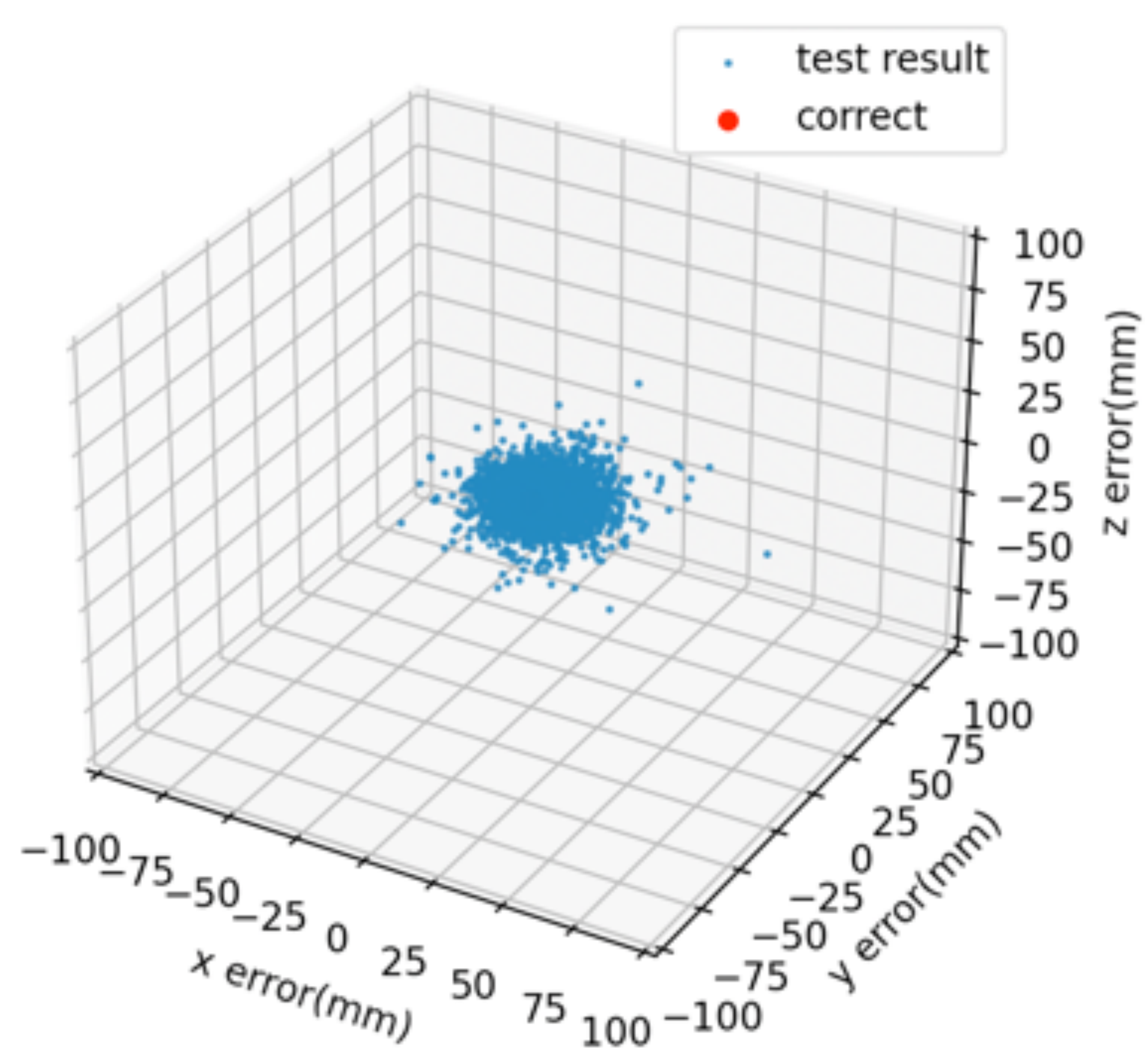
- Of the 15,000 joint angle and End-Effector position coordinate data of the manipulator published in Kaggle named 'Robot Kinematics Dataset', 12,000 are trained, and 3,000 are used for testing.



Assemble the manipulator used as data

IV Conclusions

- The learning result was judged with the data that was not used for learning 3000 out of 15000 total data. On average, an error of 10mm occurred. In the graph below, the 0,0,0 coordinates are the correct answer
  - Average x coordinate error : About 10.07mm
  - Average y coordinate error : About 10.41mm
  - Average z coordinate error : About 8.74mm



Computation Speed Comparison (sec)