Force/torque sensor calibration method by using deep-learning

Conference Paper · June 2017

DOI: 10.1109/URAI.2017.7992824

Uikyum Kim
Ajou University

Won Suk You
Wavemaker Labs

Hyouk Ryeol Choi
Sungkyunkwan University

2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)
June 28 - July 1, 2017 at Maison Glad Jeju, Jeju, Korea

Force/Torque Sensor Calibration Method by Using Deep-Learning

Hyun Seok Oh, Gitae Kang, Uikyum Kim, Joon Kyue Seo, Won Suk You and Hyouk Ryeol Choi*
School of Mechanical Engineering, Sungkyunkwan University, Suwon, Korea
(Tel : +82-31-290-7926; E-mail: hrchoi@me.skku.ac.kr, * corresponding author)

***Abstract*—The force/torque sensor is an important tool that gives a robot an ability to interact with their usage environments. Calibration is essential for these force/torque sensors to convert the raw sensor values to accurate forces and torques. However, in practice, the multi-axis force/torque sensor requires complex multi-step data processing, because of the coupling effects and nonlinearity of sensors. Moreover, accuracy is not guaranteed. To solve this problem, we propose an accurate force/torque sensor calibration method that can calibrate the sensor in single step by using deep-learning algorithm, and introduce the method for modeling the DNN(deep neural network) used in this calibration process. In addition, we also explain some tricks for learning, and then verify the calibration results through several experiments.**

***Keywords*—Force/Torque, Sensors, Calibration, Deep Learning, Deep Neural Network, Coupling Effect.**

1. Introduction

In order for the robot to physically interact with their surrounding environments, it is reasonable to get the human-like sensing system. Especially, force/torque sensors which can quantitatively measure force and torque acting on the robot is the crucial of the robot sensing system. Actually, force/torque sensors have already been applied to improve the performance of many robots [1],[2]. To measure the n-axis(≤ 6) force and torque, the sensor must be capable of producing n-dimensional outputs structurally, and calibration is essential to convert these data into useful values. In general, force/torque sensor calibration is the process of mapping raw sensor data to each force and torque. In some cases, calibration is conducted and verified only for each individual axis force value to evaluate the developed sensor [3]. However, in practice, there is uncertainty due to the coupling effects when multiple forces are applied simultaneously. For this reason, it is necessary to perform complicated data processing processes at several steps. But, there is still a problem of low accracy [4].

In recent years, deep-learning has been actively studied in the field of computer science. The powerful characteristics of this algorithm is that nonlinear learning is possible by training single DNN model including a nonlinear activation function [5],[6]. Typically, one of activation function called ReLU(Rectified Linear Unit) conditionally activate or not, which gives both nonlinearity to the DNN model and resolve the gradient vanishing problem. If these characteristics are used for sensor calibration, it is possible to calibrate not only approximate correlation between the input and desired output data but also accurate nonlinear characteristics through learning of an only single DNN(Deep-Neural-Network).

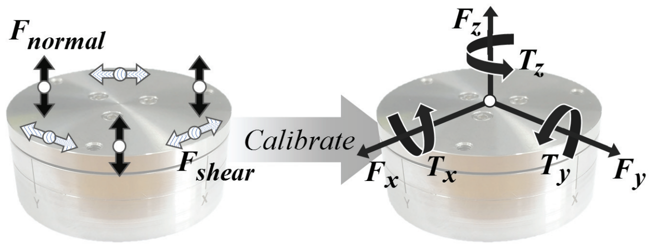

Fig. 1: Shematics for force/torque sensor calibration[8].

In this paper, we propose a novel and accurate force/torque sensor calibration method by using one of machine-learning, called deep-learning. In order to efficiently train the neural-network model for calibration, we divide the model into two parts. The previous 7-hidden layers are made to produce linear 6-dimensional data, and last hidden layer consisted of a single 6×6 linear matrix to convert linear data from previous layer to accurate force and torque. Training dataset is also important. For avoiding coupling effects directly, we use some training data which are raw sensor data and the desired output value at the time when the multi-axis force is applied to the sensor simultaneously. To verify that our method solves the coupling effect, the calibrated sensor value and theoretical value are compared when 2-axis forces of the *xy*, *xz* and *yz* planes are applied to the sensor harmoniously.

This paper is organized as follows: The fundamental principle and the overall calibration strategy is introduced in Section 2. Evaluations and discussions about the trained model are explained in Section 3. Finally, conclusions are given in Section 4.

2. Calibration Method

2.1. Basic Principle

A. Force/Torque Sensor Calibration: For the force/torque sensor to have the capability to measure the n-dimensional forces and torques through calibration, it is first necessary that sensors outputs have at least n-dimensional variation. To achieve this, some researchers developed a four degree of freedom force/torque sensor using 16-strain gauges to measure the deform of sensors cross beam [7]. In Figure 1, schematic for force/torque sensor calibration is described. The sensor used in this paper measures six capacitances to measure three normal deformation and three shear deformation of the sensors deformable part. the sensor structurally and mechanically generates a 6-dimensional capacitance data, and using these 6-capacitances, can estimate the 6-axis forces and torques [8].

Normally, assuming that m-dimensional raw data variation is linear and independent of the forces exerted on each axis of the sensor, calibration is performed using a $m \times n$-dimensional linear transformation matrix. The expression of transformation to convert the raw data from the sensor into force and torque components can be explained as Eq. (1).

$$\mathbf{F} = \mathbf{T}_c \cdot \mathbf{C} \tag{1}$$

$$Min\ \text{Err}^2 = (\mathbf{F_T} - \mathbf{F})^2 \tag{2}$$

where $\mathbf{F} \in \mathbf{R}^{6\times1}$ is estimated force and torque values and and $\mathbf{F}_T \in \mathbf{R}^{6\times1}$ is theoretically obtained reference data. $\mathbf{T}_c \in \mathbf{R}|^{6\times6}$ is linear transformation matrix that convert raw capacitance data into $\mathbf{F}$. $\mathbf{C} \in \mathbf{R}^{6\times1}$ is a measured capacitance data vector from six cells of the force/torque sensor, To get this optimized $\mathbf{T}_c$ matrix, basically least-squares method is applied as shown Eq. (2). If the sensor has the ability to output 6-dimensional linear data independently for multiple forces, it is possible to calibrate through a single conversion operation as described above.

However, unlike an ideal condition, uncertainty and nonlinearity increase due to not only coupling effects but also type of data in actual use. Summaries about the factors that affect sensor values are as follows. In the first, when get training data, changes of sensors cell values for each axis force are independently used without considering complex and various forces. Therefore, after calibration, if the multi-axis force is applied at the sensor, an error occurs due to an unexpected sensor value change. In the second, a type of data also affects the sensor error. The sensor used in this paper using capacitive type data. Capacitive type force/torque sensor is very sensitive, but there is some nonlinearity of the data fundamentally. For the accurate calibration, we also need to consider about this property.

B. Deep-Learning Algorithm: Deep-learning can learn data representation through multiple levels of abstraction using a computational model that consist of multiple processing layers. This has dramatically improved state-of-the-art technology in many areas such as natural language recognition, computer vision system and many others. The two most remarkable features of deep-learning that are noted in this paper are as follows. One is to discover organic structure in large dataset by using backpropagation algorithm. The backpropagation algorithm tunes a number of weights distributed in the model to reveal the complexity of the dataset during training. The other is that nonlinear correlation learning is possible without gradient vanishing by using nonlinear activation function. we use an activation function called ReLU which gives the model nonlinearity. The ReLU function is expressed in Eq. (3).

$$f(x) = \begin{cases} x & if \quad x > 0 \\ 0 & otherwise \end{cases} \tag{3}$$

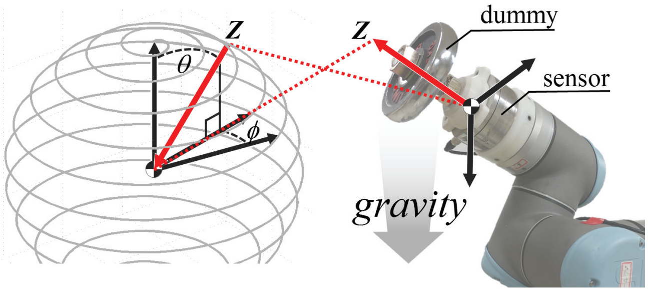

Fig. 2: Spherical helix path to get uniform oriented force vectors and experiment setup.

2.2. Training Data

A. Getting Training Data: The greatest advantage of the proposed calibration method is to eliminate the coupling effects by learning the nonlinearities that occur when the multiple axes forces are acting simultaneously. To achieve this, using the training data having force vectors with as possible as various orientation based on the sensor coordinate system. If the raw and reference data when the multi-axis force action on the sensor is used as training dataset, the coupling effect can be solved fundamentally. This training data can be obtained using robotic arm having 6-DOF(degree-of-freedom) and weight of dummy with constant mass and center of gravity [9].

In 3-dimensional space, rigid bodies generally have 6-DOF. Therefore, the 6-DOF robot is used to make the sensor have specific 3-DOF position and 3-DOF orientation each. 'UR10' is used for getting these orientations and positions, and the force/torque sensor located on end of the last link of the robot. We also use the gravity to exert the forces to the sensor, and the gravity acts on the dummy mounted with the sensor in series. The key point is to obtain the uniform and various orientations that can represent all orientations. If robots can make uniform orientations continuously, gravity will act on the dummy so that uniform and various force vectors can be input to the sensor. In Fig. 2, to define the 3-DOF position of the sensor on the robots end point, the z-axis of the sensor coordinate system always point to the center of the sphere as the left hand side. The origin always on the plane of the virtual sphere and moves along the spherical helix path. for the remaining 3-DOF orientation, using the transformation matrix as given in Eq. (4).

$$\begin{gathered} {}^S_0\mathbf{R}(\theta) = \mathbf{R}_z(-\phi) \cdot \mathbf{R}_x(\theta) \cdot \mathbf{R}_z(-\phi) \\ \phi = \alpha(\frac{\pi}{2} - \theta) \end{gathered} \tag{4}$$

where ${}^S_0\mathbf{R}(\theta)$ is the 3×3 rotation matrix to represent orientation of the sensor coordinate system. α is a constant related to the number of revolution about the helix path.

The teaching data to calibrate can be obtained theoretically through kinematic calculation of the robot arm. The equation can be expressed as Eq. (5).

$$\begin{gathered} \mathbf{F}_{sensor} = {}^S_0\mathbf{R}^T \cdot \mathbf{W} \\ \mathbf{T}_{sensor} = \mathbf{r}_{cog} \times \mathbf{F}_{sensor} \end{gathered} \tag{5}$$

where $\mathbf{W}$ is weight vector can be converts into $\mathbf{F}_{sensor}$ by mutiplying the rotation matrix. Torque can be estimate by using cross product between center of gravity vector about dummy and force vector about sensor coordinate system.

In Figure 3, the obtained training data is shown. Only 3-axis of 6-axis force and torque data are expressed, and only 3-raw data of the capacitances measured by the sensor are shown. The forces are theoretically obtained teaching data by caculating the kinematic of the robot, and the capacitances are raw data output from sensor.

B. Characteristics of Training Data: To ensure how uniform and various the orientation of the input force vectors are, distribution of obtained force vectors are displayed in Fig. 4. It can be seen that the force vectors to be used for learning are variously and uniformly distributed. Since the torque value depends on the force value, it is only necessary to consider the 3-axis force data.

2.3. *Test Data*

To evaluate our calibration, trained model should be checked through the both training and test data. To make sure that proposed method solves the coupling effect or not, the following method was applied. Some researchers have tried to verify the calibration for a single axis force, but, this method does not ensure if the coupling effect has been resolved. For this reason, of the 3-axis forces, 1-axis force is constrained and the 2-axis forces are harmonically applied to the sensor. The tendency of the applied force is shown in Fig. 6(b). At the same time, just two kinds forces are applied to sensor.

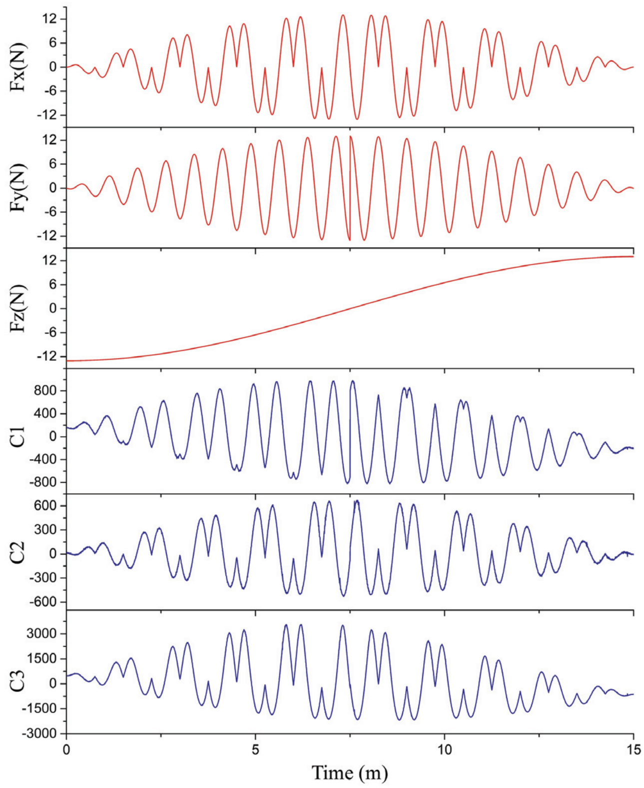

Fig. 3: F_x, F_y and F_z are the theoretical forces input to the sensor obtained through the kinematic analysis of the robot arm. C1, C2 and C3 are capacitance value obtained from the sensor.

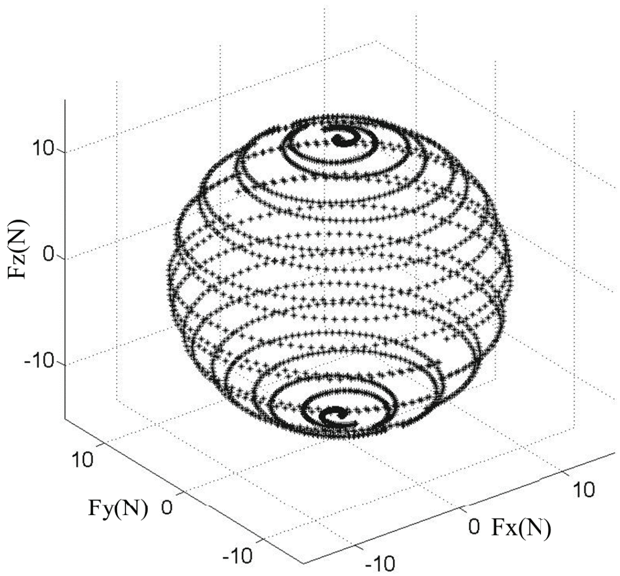

Fig. 4: Distribution about force vectors of training data.

2.4. *Network Design*

In order to calibrate the force/torque sensor using deep-learning, we modeled the deep-neural-network as shown in Fig. 5. The concept we are considering when modelling a network is to separate the roles of layer. Eight hidden layers are designed between input layer and output layer. The input layer is normalized capacitance value of the sensor and the output layer should be the same as desired force/torque. The first eight hidden layers transform the output of the sensor into linear six-dimensional data. The 7th hidden layer uses the hyperbolic tangent function as the activation function when passing data to The 8th hidden layer which serves to center the mean value to '0' and level the absolute value to '1'. In

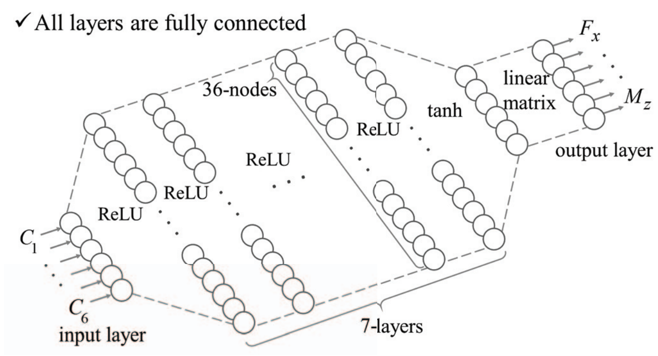

Fig. 5: Deep-neural-network for calibration.

the process if calculating data from the 8th hidden later to the output layer, a 6×6 linear matrix is used as the transformation matrix. All layers are fully connected to get the complex correlation of data. The each of first seven hidden layers has 36 nodes.

2.5. Training

For calibration, the deep-neural-network model in Fig. 5 should be trained. Several conditions were carried out in order for the model to be trained. Used training data in Fig. 3 are 4000 sampled values while sensored last link of robot passes the spherical helix path. Throughout training, we use a batch size of 4000(full-batch), Adam optimizer. Bottom three graphs In Fig. 3, each output of the sensor has different arrange due to the difference of each cells sensitivity. Considering these characteristics, each cell output values are divided by its maximum value to normalize. Loss function is described in Eq.(6).

$$\text{loss} = \left|Fx - Fx'\right|^2 + \left|Fy - Fy'\right|^2 + \left|Fz - Fz'\right|^2 + \frac{F_{max}}{T_{max}}\left(\left|Tx - Tx'\right|^2 + \left|Ty - Ty'\right|^2 + \left|Tz - Tz'\right|^2\right) \tag{6}$$

where F_x, F_y, F_z, T_x, T_y and T_z are teaching values like upper three graphs in Fig. 3. $F_x^{'}$, $F_y^{'}$, $F_z^{'}$, $T_x^{'}$, $T_y^{'}$ and $T_z^{'}$ are the estimated ones from the DNN. In Loss function, we divide the torque error term by $T_{\max}/F_{\max}$. Because of the maximum values of torque and force are different due to the dummy used in obtaining training data has a weight of 1.3kg and a center of gravity of [0 0 0.087](mm).

We use the 3-step drop-out concept to learn the model evenly. Drop-out can make the model avoid the overfitting which is fatal in learning due to arouse the error in test data through losing the general feature between input and output data. But, in this paper, the last hidden layer of DNN give the generalized relation between input and output by using linear transformation matrix. Therefore, using dropout rather increase the error. Our learning configure is in TABLE 1.

TABLE 1: Training step

Epochs	Learning rate	Drop-out rate
1∼2000	0.01	0.4
2001∼4000	0.01	0.2
4001∼8000	0.01	0.0

In these method, dropout-rate gradually decreases step by step and becomes ‘0’ in the third step. This method initially sets the direction of uniform learning in the model. Finally, with the ‘0’ dropout rate, we get the accuracy trained model. For these specific values, some parameters are considered such as learning rate, epochs, hidden layers thickness, number of node and etc. In Section 3, we mention about optimize process briefly.

3. Optimization and Evaluation

3.1. Optimization of Deep-neural-network

In order to model the DNN in Section 2.3, we consider some constraints and parameters about training and model. There are too many factors that affect the results of deep learning. Therefore, after the basic model as shown in Fig. 5 was selected through trial and error, only two parameters were considered. One is whether the dropout is applied or not, the other is how many layers to apply hyperbolic tangent activation function to. The layer to be applied the hyperbolic tangent activation function is the 7th or 7th, 8th, and the dropout is the 3-step dropout mentioned in Section 2.4. In order to find the optimal condition, we compare the loss about both training data and test data. The optimized result in TABLE 2. Case 3 has the smallest loss with training data but, case 1 has the smallest loss with test data. To avoid overfitting and reduce the coupling effect, loss of test data is more important. That is the reason why we choose the model of case 1.

TABLE 2: Loss about each case

case	tanh	drop-out	loss(train)	loss(test)
1	8th layer	O	0.002826	0.234962
2	8th layer	X	0.003002	0.249229
3	7, 8th layer	O	0.002462	0.255885
4	7, 8th layer	X	0.002508	0.259917

3.2. Experimental evaluation

A. Result with training data: As shown in Figure 6(a), the proposed method very accurate at the training model. Dotted black line is the reference data and the red line is the estimated data through DNN. Suprisingly, the error of the blue line is almost ‘0’. But, the verification of the trained model requires the accuracy about the test data. Therefore, verifying with the test data is proposed in Section 2.3. To get the 2-axis force input, the robot uses the dummy used to obtain the learning data and goes through a different path. Beyond this path, 2-dimensional force is input to the sensor due to the weight of the dummy.

The z-axis torque should always be ‘0’. Because, the dummy used to get the data has the center of gravity which is mathches the z-axis of sensor. Therefore, Tz becomes ‘0’ by Eq. (5). But, if we use a weight dummy with an offset center of gravity for the z-axis, then the non-zero Tz can also be obtained.

B. Results with test data: The result of the calibration on the test data is shown in Fig. 6(b). We divide the test data into 3-phase to distinguish the exerted force plane. To analyze the calibration results, data was analyzed on a phase-by-phase basis and discussions were conducted.

.

TABLE 3 shows the errors about Figure 6(b). In the phase 1, x-y plane force is applied the sensor, phase 2 show the error about x-z plane force and phase 3 shows about y-z plane

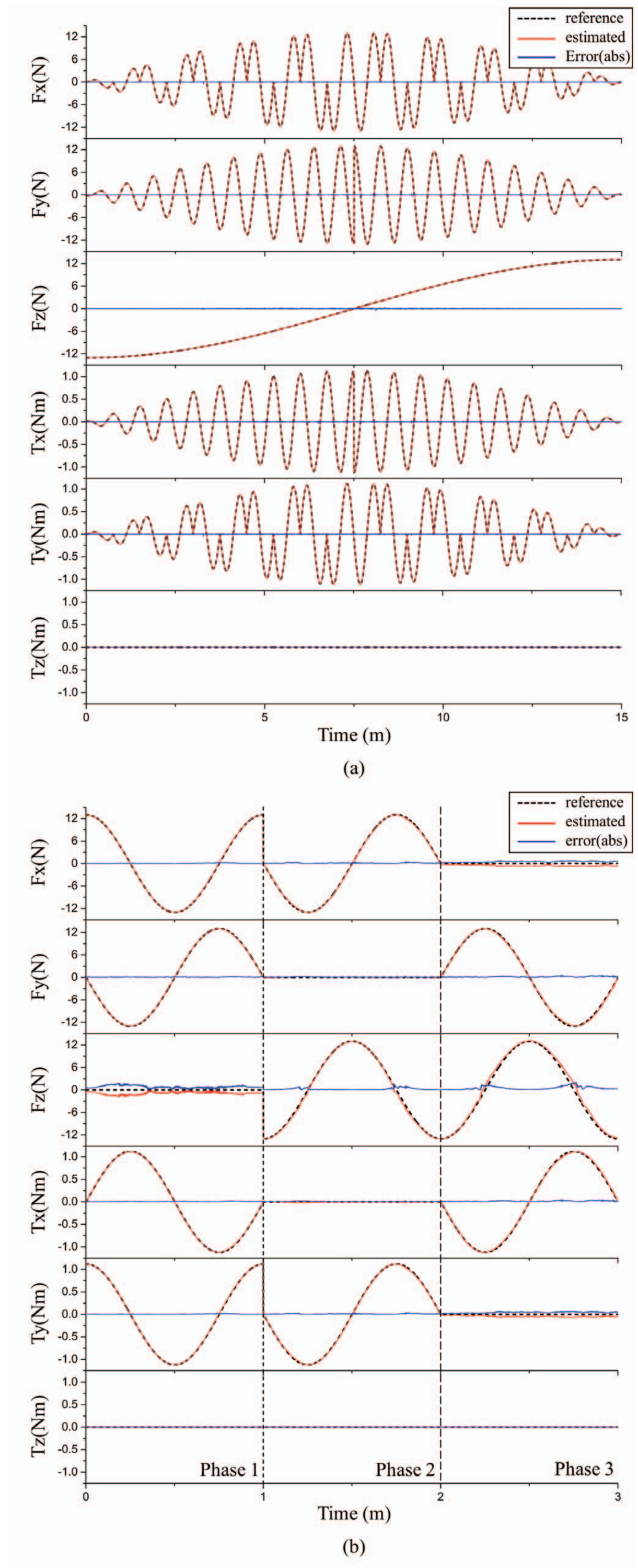

Fig. 6: (a) Time domain response about training dataset. Dotted black is the reference force/torque, red line is estimated sensor data and blue line is the absolute error. (b) Difference between result data and reference data. To check the coupling effect, each force and torque response divided into 3-phase.

TABLE 3: Error for each axis

error type		phase1	phase2	phase3
Fx	max	0.3399	0.5083	0.7408
	mean	0.0636	0.1145	0.4927
Fy	max	0.2363	0.2859	0.5687
	mean	0.0718	0.0605	0.1621
Fz	max	1.8833	1.6586	2.0226
	mean	0.8545	0.2158	0.5225
Tx	max	0.0212	0.0246	0.0527
	mean	0.0063	0.0052	0.0140
Ty	max	0.0292	0.0403	0.0639
	mean	0.0055	0.0099	0.0424
Tz	max	0.0000	0.0000	0.0000
	mean	0.0000	0.0000	0.0000

force. Fz has the largist error both of maximun and mean values in all phases. This is because the uncertainty of Fz is greatest due to the characteristic of the sensor. As shown in Fig. 6(b), for test values other than the z-axis force, satisfactory results are obtained. These results means the DNN can be learn complex corealation between input sensor values and output force/torque values.

One more experiment was conducted to confirm the improvement of sensor performance through the proposed calibration method. One of the two methods to compare is the proposed calibration method and the other is the linear transformation method described in Eq. (2).

As shown in Figure 7(a), when the calibration is performed

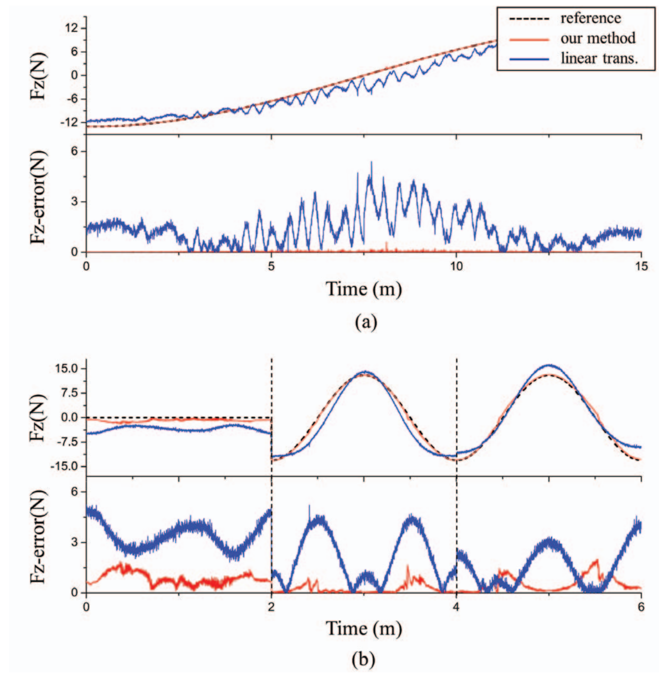

Fig. 7: (a) Comparison calibration result about training data between our method and linear transformation. (b) Comparison between ours and linear transformation about test data.

using the linear transformation method, it can be seen that the error of Fz occurs in accordance with the change tendency of x and y axis forces. This result is referred to as the coupling effect mentioned above. But, the results of the proposed method show that the coupling effect is largely solved. These results indicate that the proposed calibration method is capable of learning not only the relationship between input and output data but also nonlinear learning. Througt these characteristics, if the DNN can be trained about more various and uniform forces, this method can brings a tremendous accuracy as shown in Fig. 6(a).

4. Conclusions

In this paper, a novel and accurate force/torque sensor calibration method by using deep-learning was proposed. In order to overcome the coupling effects, which is a chronic problem existing in the force/torque sensor issue, data of various orientations are obtained and the data are learned non-linearly by using a deep-learning algorithm. We also designed a model to improve learning efficiency. The models used were divided into roles, not single types of layers. This learning method shows very high accuracy for training data and much higher accuracy than linear transformation method in test data. But, if learning about test data is implemented, it will surely give better results. because occured error in Fig. 6(b) is due to the diversity of learning data is limited to specific conditions we made. In future work, we will explore ways to obtain more diverse learning data through a simple method and to obtain a more optimized DNN model.

Acknowledgement

This research was conducted under the robot industry cluster construct program, which is funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea)

References

[1] J. Butterfass, M. Grebenstein, H. Liu and G. Hirzinger, DLR-Hand II: Next Generation of a Dextrous Robot Hand, International Conference on Robotics & Automation, Soeul, 2001.

[2] U. Kim, D. -H. Lee, W. Yoon, B. Hannaford and H. R. Choi, Force Sensor Integrated Surgical Forceps for Minimally Invasive Robotic Surgery, IEEE Transactions on Robotics, Vol. 31, No. 5, pp. 1214-1224, 2015.

[3] Q. Liang, D. Zhang, Q. Song, Y. Ge, H. Cao and Y. Ge, Design and fabrication of a six-dimensional wrist force/torque sensor based on E-type membranes compared to cross beams, Measurement, Vol. 10, No. 1, pp. 87-97, 1997.

[4] T. Lu, G. C. Lin and J. R. He, Neural-Network-Based 3D Force/Torque Sensor Calibration for Robot Applications, Engineering Applications of Artificial Intelligence, Vol. 43, No. 10, pp. 1702-1719, 2010.

[5] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, Nature, Vol. 521, pp. 436-444, 2015.

[6] V. Nair and G. Hinton, Rectified Linear Units Improve Restricted Boltzman Machines, International Conference on Machine Learning, Haifa, 2010.

[7] A. Song, J. Wu, G. Qin and W. Huang, A novel self-decoupled four degree-of-freedom wrist force/torque sensor, Measurement, Vol. 40, No. 9-10, pp. 883.891, 2007.

[8] U. Kim, D. -H. Lee, Y. B. Kim, D. -Y. Seok and H. R. Choi, A Novel 6-Axis Force/Torque Sensor for Robotic Applications, IEEE/ASME Transactions on Mechatronics, (2016). [Online]. http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7784722

[9] H. S. Oh, U. Kim, G. Kang, W. S. You, H. R. Choi, Calibration method for a force/torque sensor based on usage environment with a Robot arm, The 12th Korea Robotics Society Annual Conference, Peongchang, 2017.