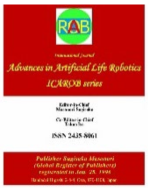
SUGISAKA MASANORI

Journal of Advances in Artificial Life Robotics
Vol. 3(1); June (2022), pp. 8–13
ON LINE ISSN 2435-8061; ISSN-L 2435-8061
https://alife-robotics.org/jallr.html



Research Article

Weight estimation for noodle products in food layout of a home replacement meal

Tomofumi Tsuji[1], Sakmongkon Chumkamon[1], Chanapol Piyavichayanon[2], Ayumu Tominaga[3], Ryusuke Fujisawa[1], Abbe Mowshowitz[4], Eiji Hayashi[1]

[1]*Department of Mechanical Information Science and Technology, Kyushu Institute of Technology, 680-4, Kawazu, Iizuka-City, Fukuoka, 820-8502, Japan*

[2]*Department of Creative Informatics, Kyushu Institute of Technology, 680-4 Kawazu, Iizuka, Fukuoka 820-8502, Japan*

[3]*Department of Creative Engineering Robotics and Mechatronics Course, National Institute of Technology Kitakyushu Colllege, 5-20-1 Shii, Kokuraminamiku, Kitakyushu, Fukuoka, 802-0985, Japan*

[4]*Department of Computer Science, The City College of New York, 160 Convent Avenue, New York, NY 10031, USA*

ARTICLE INFO

Article History

Received 8 November 2021
Accepted 27 June 2022

Keywords

FA robots
Image processing
Noodle products
Weight estimation
Deep learning
General dataset

ABSTRACT

There is a growing demand for automation by robots in the home replacement meal industry due to labor shortages in food factories and from the perspective of the SDGs[1]. In this research, we are developing autonomous work robots that can perform home replacement meal tasks and developing the technology for industrial food automation using Artificial Intelligence (AI) to improve productivity, security, and safety. In this paper, we perform weight estimation of the served object to identify the amount of spaghetti grasped by the robot. We created our dataset of spaghetti used for weight estimation. Spaghetti of varying weight is in different types of containers, placed at random positions in the robot workspace. The proposed model is shown to estimate the weight of spaghetti with an error of at most 10%.

1. Introduction

In recent years, there has been an increasing demand for robot automation to improve productivity in the food layout of lunchboxes and prepared foods in the Japanese home replacement meal industry. In this research, we are developing autonomous work robots that can perform home replacement meal tasks

In a previous study, we developed a system that can adequately grasp and serve solidified foods such as fried chicken and rice balls. However, in addition to solid foods, noodle products such as spaghetti are used as ingredients for lunch boxes and prepared foods in the foodservice industry. In addition, it is necessary to consider the quantitative properties of noodle products when placing them in containers. To realize quantitative serving of noodle products, robotic functions developed in previous studies need to be implemented.

In this research, we perform weight estimation of the served object to identify the amount of spaghetti grasped by the robot. We also propose a deep learning method using RGB-D cameras for weight estimation and describe its validation and evaluation.

Corresponding author's E-mail: tsuji.tomofumi552@mail.kyutech.jp, m-san@mmcs.mse.kyutech.ac.jp, iyavichayanon.chanapol402@mail.kyutech.jp, tominaga@kct.ac.jp, amowshowitz@ccny.cuny.edu, fujisawa@ces.kyutech.ac.jp, haya@mse.kyutech.ac.jp
http://www.kyutech.ac.jp/

2. System Overview

2.1. *System configuration*

The system configuration of this research is shown in Fig. 1. In this system, a robot grasps spaghetti, arranges it in a specific container, and estimates its weight. The system consists of a visualization processing unit and a robot control unit. First, the image data from the RGB-D camera mounted on the robot is collected and analyzed. The visualization process includes real-time object recognition and 3D processing of food images. That computer also performs weight estimation after the spaghetti is served. A computer based system for Robotics Motion controller receives the information from vision processing and sends the motion planning signal to the controller, which enables the robot to grasp and serve the food.

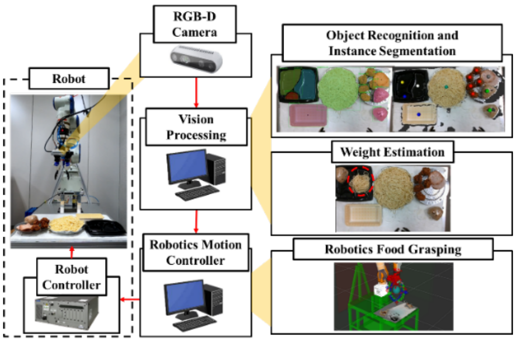


Fig. 1. System configuration diagram

2.2. *Weight Estimation System in Spaghetti*

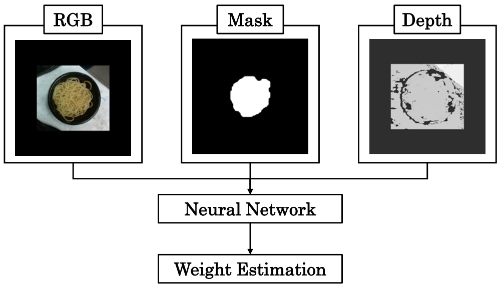


Fig. 2. Weight estimation system configuration diagram

The weight of the spaghetti is estimated by using the information from the RGB-D camera as well as the object recognition system. The system shown in Fig. 2 receives RGB, Mask, and Depth data from an RGB-D camera as input, and estimates the weight using a neural network. By using the three data as feature values, when spaghetti is placed in a container, the system can identify the object for weight estimation by RGB and Mask, and it can also specify the surface shape of the object in three dimensions by Depth. The goal of the neural network is to capture the surface shape of the object and estimate the density of the entire object, including the container.

3. Robot configuration

The appearance of the robot in this study is shown in Fig. 3. The robot is a 7-axis vertically articulated robot that has a shape like a human arm and can move with a high degree of freedom because it is required to perform the same work as a human in a food factory. The robot is equipped with an RGB-D camera, an end-effector, and a force torque sensor. The robot is equipped with two types of grippers to grasp the object. The tongue gripper is used for grasping foods with rough surfaces such as fried foods, rice balls, and spaghetti. Vacuum gripper is for gripping objects with a smooth surface, such as ham and containers. The gripper can be selected instantly by the servo motor according to the target food.

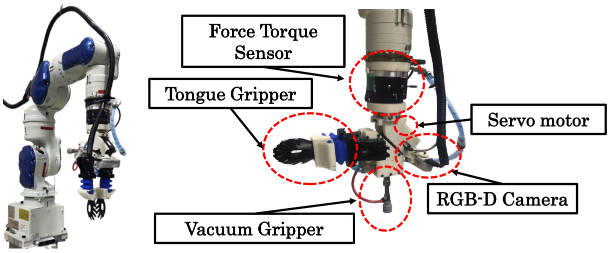


Fig. 3. Appearance of the robot

4. Building Deep Learning Models

Since the training data for our system are RGB, Mask, and Depth, we need to consider the complexity of the neural network and the reduction of computational complexity for a large amount of data. Therefore, we used the idea of ResNeXt[2], a deep learning model, to build a model that enables weight estimation. ResNeXt is an extension of ResNet[3] (Residual Network), which solves the problem of gradient loss due to too many layers in a neural network by using a residual block consisting of a convolutional layer input and a Shortcut Connection. The Shortcut Connection outputs residuals across layers to prevent gradient decay. In addition, ResNeXt extends the residual blocks of ResNet to branch out a new dimension of cardinality, which improves the expressive power. Fig. 4 shows the structure of the deep learning model we constructed. For each input RGB, Mask, and Depth, we designed a CNN convolutional layer, a maximum pooling layer, and a Basic Block (Cardinality=32) which is the basis of ResNeXt. The structure of the Basic Block is shown in Fig. 5. The Basic

Block is designed again by fusing these three inputs, and then the weight of spaghetti is predicted by the average pooling layer and the Fully Connected Layer.

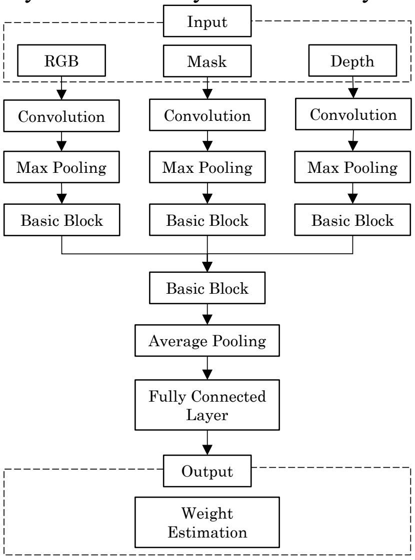


Fig. 4. Weight estimation system configuration diagram

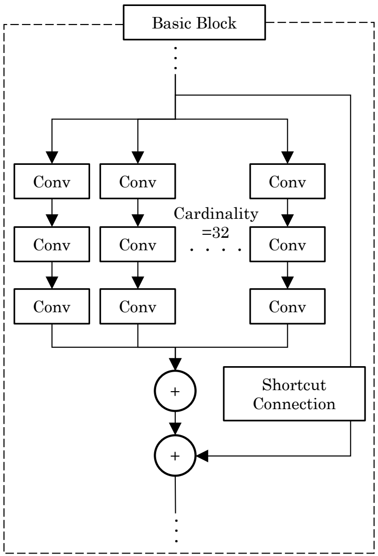


Fig. 5. Configuration of Basic Block

The activation function immediately after the convolutional layer is an ELU (Exponential Linear Unit), and the relationship between the output $f(x)$ and the input x of the ELU is shown in Eq. (1). in the ELU, if the input value is negative, the output value also returns a negative value. In ELU, if the input value is negative, the output value will also return a negative value, thus speeding up the convergence of the loss and improving the stability of the signal.

When training this deep learning model, Smooth L1 Loss was used as the loss function to indicate the training accuracy. Eq. (2) and (3) show the results. The parameter beta was set to 1. The optimization method used to update the weight parameters in deep learning is Adam (A Method for Stochastic Optimization)[4], which allows efficient exploration of the parameter space. momentum by using the average of the gradient. Equations (4), (5), and (6) show the equations for updating the weights w. v_{t-1}, s_{t-1} : the value after moving average at the previous time, β_1: hyperparameter from 0 to 1, G: value at the current time, α: learning rate, ε: value to prevent zero division.

$$f(x) = \begin{cases} \alpha(e^x - 1), & x \leq 0 \\ x, & x > 0 \end{cases} \quad (\alpha > 0) \tag{1}$$

$$\mathrm{loss}(x, y) = \frac{1}{n}\sum_i z_i \tag{2}$$

$$z_i = \begin{cases} 0.5(x_i - y_i)^2/beta, & if\,|x_i - y_i| < beta \\ |x_i - y_i| - 0.5 * beta, & \text{otherwise} \end{cases} \tag{3}$$

$$v_t = \beta_1 v_{t-1} + (1 - \beta_1)G \tag{4}$$

$$s_t = \beta_2 s_{t-1} + (1 - \beta_2)G^2 \tag{5}$$

$$w_t = w_{t-1} + \alpha\frac{v_t}{\sqrt{s_t + \varepsilon}} \tag{6}$$

5. Experiment

5.1. *Dataset*

We created a dataset with weight as the correct answer label to develop a weight estimation system. The dataset consisted of one pair as RGB, Mask, and Depth for objects whose weight was measured in 0.1g increments. Since spaghetti is irregularly shaped, we used six different containers of different shapes and depths to obtain the data. The weight of the containers ranged from 49.8g to 200.3g, considering the maximum grasping capacity of the robot and the spaghetti in a medium meal. The dataset consisted of 3,806 pairs with respect to weight and container.

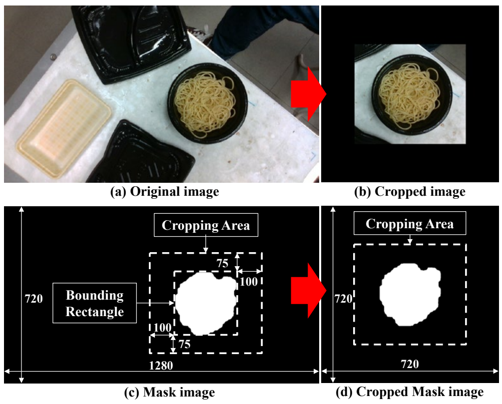


Fig. 6. Feature suppression by image cropping

The data is also processed for cropping (Fig. 6). The required feature are spaghetti, the container, and the areas around the container with different heights. As shown in Figures 6 (c) and 6 (d), the short form of the Mask was held and enlarged 100 pixels to the left and right and 75 pixels to the top and bottom to include the container and its background. An example of the dataset is shown in Fig. 7. The dataset was randomly divided into 80% (3044 pairs) training data, 10% (381 pairs) validation data, and 10% (381 pairs) test data. This dataset is a general-purpose dataset that considers each data set independent of the surrounding background, camera position, and height.

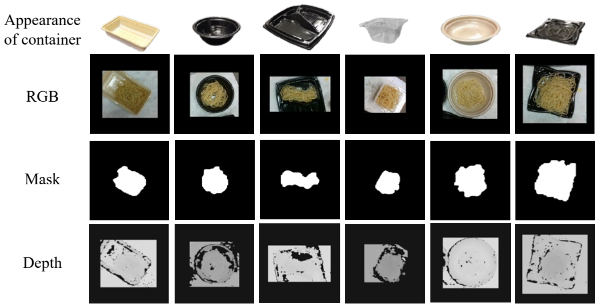


Fig. 7. Example of the dataset for the spaghetti weight estimation, including the various types of containers.

5.2. *Result*

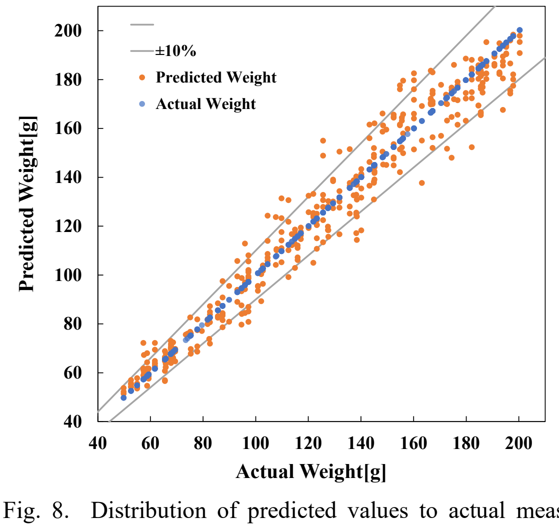


Fig. 8. Distribution of predicted values to actual measured values

Weight estimation with test data was performed using data trained by a deep learning model. The number of training sessions was set to 3,000, the learning rate to 0.001, and the batch size to 16. The results of the predictions against the actual measurements are shown in Fig. 8. Histogram of percent error and Frequency of percent error is shown in Fig. 9 and Table 1, respectively. The mean absolute error MAE[g], the mean absolute percent error MAPE[%], and the standard deviation, maximum error, and minimum error for each unit are shown in Table 2. Eq. (7) and (8) are used to calculate MAE and MAPE, respectively.

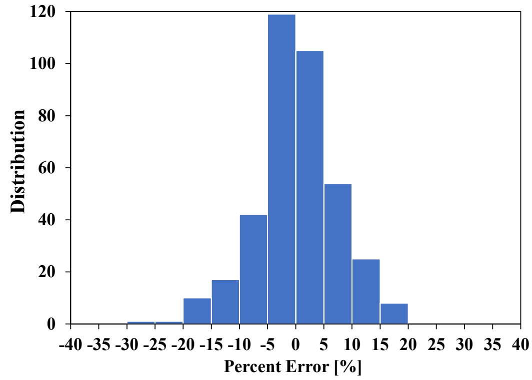


Fig. 9. Histogram of percent error

Table 1. Frequency of percent error

Percent Error [%]	Distribution
-30 ~ -25	1
-25 ~ -20	1
-20 ~ -15	10
-15 ~ -10	17
-10 ~ -5	42
-5 ~ 0	119
0 ~ 5	105
5 ~ 10	54
10 ~ 15	25
15 ~ 20	8
20 ~ 25	0
25 ~ 30	0

$$MAE = \frac{1}{N}\sum |A_t - F_t| \quad (7)$$

$$MAPE = \frac{1}{N}\sum \frac{|A_t - F_t|}{A_t} \quad (8)$$

Table 2. Evaluation index of predicted values

Parameters	Error [g]	Percent Error [%]
Mean Absolute	6.65	5.38
Standard Deviation	6.02	4.52
Maximum Error	29.7	26.1
Minimum Error	0.00961	0.0112

5.3. *Discussion*

In this experiment, from Fig. 7 and Table 2, the MEPE and standard deviations are 5.38% and 4.52%, respectively, which means that the weight estimation can be done with an accuracy of ±10% percent error. The histogram in Fig.8 also shows that -5%~5% is the most significant percentage, and the percentage with large errors is small, which indicates that the learning and data are valid. Since there was no consistency in the data with relatively large errors, it is expected that the accuracy can be improved by increasing the number of such data as training data and by reviewing the training model and parameter design.

6. Conclusion

In this study, we developed a system that can estimate the weight of noodle products. This system enables the identification of the grasping amount of noodle products. The versatility of the dataset and the balance between the complexity of the neural network and the amount of computation are examined theoretically, and as a result, it is possible to estimate the weight of spaghetti with an accuracy of generally less than 10% error.

References

1. World Health Organization. World health statistics 2016: monitoring health for the SDGs sustainable development goals. World Health Organization, 2016.
2. S.Xie, R.Girshick, P. Doller, Z.Tu, K.He, UC San Diego. “Aggregated Residual Transformations for Deep Neural Networks.” In CVPR, 2017.
3. K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition.” In CVPR, 2016.
4. Diederik P. Kingma and Jimmy Lei Ba. Adam : A method for stochastic optimization. 2014.

Authors Introduction

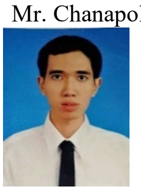
Mr. Tomofumi Tsuji

He received bachelor degree in Engineering in 2021 from mechanical system engineering, Kyushu Institute of Technology in Japan. He is currently a Master student at Kyushu Institute of Technology and conducts research at Hayashi Laboratory.

Dr. Sakmongkon Chumkamon

Dr. Sakmongkon Chumkamon received Doctor of Engineering degree from Kyushu Institute of Technology in 2017. He was a postdoctoral researcher at Guangdong University of Technology in 2017-2019. Presently he is a postdoctoral researcher in Kyushu Institute of Technology since 2019. His research interests include factory automation robots and social robots.

Mr. Chanapol Piyavichayanon

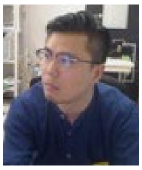


He received bachelor degree in Engineering in 2020 from Mechanical Engineering, Chulalongkorn University in Japan. He is currently a Master student at Kyushu Institute of Technology and conducts research at Koga Laboratory.

Projected Assist. Prof. Ayumu Tominaga

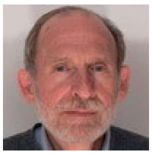
Projected Assist. Prof. Ayumu Tominaga is a professor in Department of Creative Engineering Robotics and Mechatronics Course at National Institute of Technology Kitakyushu College. He received the Ph.D. (Dr. Eng.) degree from Kyushu Institute of Technology in 2021. His research interests include Intelligent mechanics, Mechanical systems and Perceptual information processing.

Assoc. Prof. Ryusuke Fujisawa



Assoc. Prof. Prof. Ryusuke Fujisawa is an associate professor in the Department of Intelligent and Control Systems at Kyushu Institute of Technology. He received the Ph.D. (Dr. Eng.) degree from The University of Electro-Communications in 2009. His research interests include Swarm intelligence, Swarm robotics and Distributed autonomous system. He is a member of Japan Society of Civil Engineers (JSCE), Information Processing Society of Japan (IPSJ), The Society of Instrument and Control Engineers (SICE) and The Robotics Society of Japan (RSJ).

Prof. Abbe Mowshowitz



Prof. Abbe Mowshowitz received the Ph.D. degree from University of Michigan in 1967. He has been professor of computer science at the City College of New York and member of the doctoral faculty at the Graduate Center of the City University of New York since 1984. His current research interests lie in two areas are organizational and managerial issues in computing, and network science. In addition to teaching and research, He has acted as consultant on the uses and impacts of information technology (especially computer networks) to a wide range of public and private organizations in North America and Europe.

Prof. Eiji Hayashi

Prof. Eiji Hayashi is a professor in the Department of Intelligent and Control Systems at Kyushu Institute of Technology. He received the Ph.D. (Dr. Eng.) degree from Waseda University in 1996. His research interests include Intelligent mechanics, Mechanical systems and Perceptual information processing. He is a member of The Institute of Electrical and Electronics Engineers (IEEE) and The Japan Society of Mechanical Engineers (JSME).