

## Research Article

# Control of a Ball Catch Robot Using Machine Learning

Shinichi Imai\*

Graduate School, Tokyo Gakugei University, 4-1-1, Nukuikita-machi, Koganei, Tokyo 184-8501, Japan

### ARTICLE INFO

#### Article History

Received 10 October 2019  
 Accepted 14 June 2020

#### Keywords

Machine learning  
 control  
 experiment  
 evaluation

### ABSTRACT

Robots, such as industrial robots, have been used in the world of industry since the 1970s. There has been particularly rapid development in the field of robots in recent years, and there has been progress in robot research in industries such as communications and automobiles. For this reason, in the near future, robots with a diverse range of applications will be required around us. In this paper, as part of foundational research on robots and artificial intelligence, we propose a method for learning ball trajectories, using machine learning, to estimate target values for the distance in which robots move. In the proposed method, we use a linear regression model for supervised learning, and validate its effectiveness through experimentation.

© 2020 The Authors. Published by Atlantis Press B.V.

This is an open access article distributed under the CC BY-NC 4.0 license (<http://creativecommons.org/licenses/by-nc/4.0/>).

## 1. INTRODUCTION

Robots, such as industrial robots, have been used in the world of industry since the 1970s. There has been particularly rapid development in the field of robots in recent years, and there has been progress in robot research in industries such as communications and automobiles [1,2]. For this reason, it is thought that robots will exist all around us in the near future. Recently, we have also seen the birth of innovation, in which new technologies, such as Internet of Things (IoT) [3,4], Artificial Intelligence (AI) [5,6], and Big Data, are being incorporated into all industries and society and lifestyles, and Society 5.0 is being promoted to resolve social issues in a way that meets the needs of each individual.

Additionally, the RoboCup [7–9] project is being conducted on the theme of soccer, using autonomous mobile robots, to promote research into the fusion of robot engineering and AI. In this project, a significant amount of research has taken place into the fusion of robot engineering and AI, but most has concerned tracking of robot movement etc., and there have been few studies focusing on the control methods to estimate the trajectory of the ball. Other reports include learning of billiards using reinforcement learning [10] and cases of AI soccer [11]. In this paper, as part of foundational research on robots and AI, we propose a method for learning ball trajectories, using machine learning, to estimate target values for the distance in which robots move.

## 2. DESIGN OF REFERENCE

Here, we explain the method for estimating target values using machine learning. The system concept diagram is shown in Figure 1.

\*Email: [shimai@u-gakugei.ac.jp](mailto:shimai@u-gakugei.ac.jp); <http://www.u-gakugei.ac.jp/>

Here, a motor for moving a keeper robot is controlled and we use feed-forward control. Additionally, “ $y$ ” is control output, “ $u$ ” is control input, and  $r$  is the target value. The target value “ $r$ ” refers to the position of the keeper robot, and the target value is estimated, using machine learning, based on the shoot angle of the attacker robot and the position of the ball.

### 2.1. Machine Learning

In the case of machine learning, we use a linear regression model with supervised learning. The linear regression model can be shown using the following formula.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p \quad (1)$$

where,  $Y$  is a response variable,  $X$  is predictor variable  $X_p$  ( $i = 1, 2, \dots, p$ ),  $\beta_0$  is a fragment,  $\beta_i$  is the coefficient of each of the predictor variables, and  $p$  is the number of predictor variables. With linear regression, the model has predictor variable coefficient and fragment group  $\{\beta_i\}_{i \in \{0, p\}}$  as parameters. Further, using the vector/index notation, the linear regression model can be expressed as follows:

$$Y = \beta X \quad (2)$$

where,

$$F = \frac{1}{2} \|Y - X\beta\|_2^2 \quad (3)$$

where,  $\|\cdot\|_2$  is the  $l_2$  norm. At this time, the least-squares estimator is defined as follows:

$$r := \arg \min_{\beta} F \quad (4)$$

where,  $r$  is the gradient of  $F$  with respect to  $\beta$

$$r = -X^T(Y - X\beta) = 0 \quad (5)$$

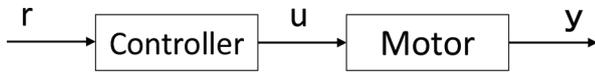


Figure 1 | Block diagram.

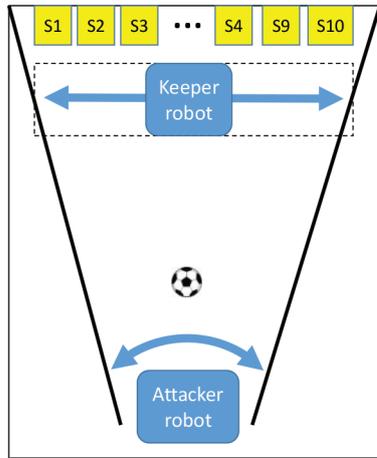


Figure 2 | Conceptual diagram of experiment.

The solution of is the least-squares estimate. Therefore,  $r$  is calculated by the following equation:

$$r = (X^T X)^{-1} X^T Y \tag{6}$$

The estimated amount obtained here is used as reference.

### 3. EXPERIMENTAL SYSTEM

Figure 2 shows the configuration of the experiment system, and Figure 3 shows the circuit diagram of the experiment device. The experiment system, as shown in Figure 3, is comprised of an attacker robot and a keeper robot. As shown in Figure 4, to detect the position of the ball, 10 touch sensors are placed at equal intervals. The keeper robot is in front of the touch sensor, and he can catch the ball. The keeper robot can only move left or right. Next, the attacker robot has a potentiometer attached, and is able to measure the shoot angle of the ball. The attacker robot moves manually. Here, it obtains the voltage of the potentiometer to detect the shoot position, and the given control quantity is provided as an instruction to the motor of the keeper. The controller, as shown in Figure 5, uses the Raspberry Pi, as well as Python, which is often used in the programming of machine learning and AI. As the Raspberry Pi is equipped with A/D conversion, it is not possible to directly incorporate an analog sensor. For this reason, this is supported by attaching an A/D converter between the analog sensor and Raspberry Pi. There is a switch used to switch between “learning mode”, in which learning is performed, and “execution mode”, in which it actually moves, and two switches for starting. The details of the hardware are shown in Table 1.

The response variable ( $Y$ ) in Subsection 2.1 expresses the angle of the attacker, and the predictor variable ( $X$ ) shows the position of the limit switch.  $\beta$  in Expression 1 is calculated by machine learning. We can estimate the position of the keeper (target value) using these two variables. Estimation of learning and target values are all

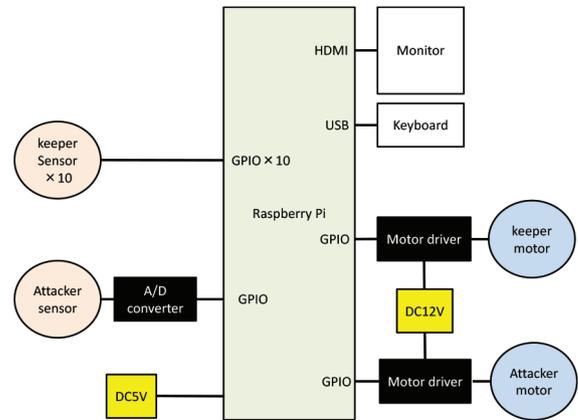


Figure 3 | Circuit.



Figure 4 | Ball detection sensor.

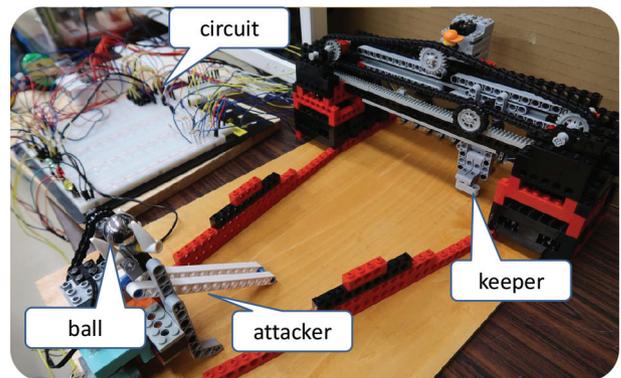


Figure 5 | Experimental system.

Table 1

Hardware	Raspberry Pi 3 Model B
Software	Python 3
Library	RPI.GPIO SciKit-learn
A/D converter	mcp3008 (10 bit 8 ch)
Sensor	Potentiometer (RV30YN40R) Microswitch (JF11210) × 10
Actuator	LEGO Mind storm RCX motor × 2
Other	Motor driver (TA7291P)

performed using the Raspberry Pi. Furthermore, the learned data and estimated target values are stored by the Raspberry Pi in SD memory in csv format.

### 4. EXPERIMENTAL RESULT

The results of learning the kicker robot angle and goal position are shown in Figures 6–8. Learning takes place five times in Figure 6, 10 times in Figure 7, and 100 times in Figure 8. We can see that, as the number of learning times increase, the more the positional relationship between the kicker robot angle and goal approaches a linear one.

The target values are estimated using this learning data, shown in Figure 9. The results of the number of learning times and the correct answer rate, in terms of catching the ball, are shown in Figure 10. Here, the ratio of catching the ball can be defined as shown in the following formula:

$$\text{Accuracy} = \frac{TP}{N} \tag{7}$$

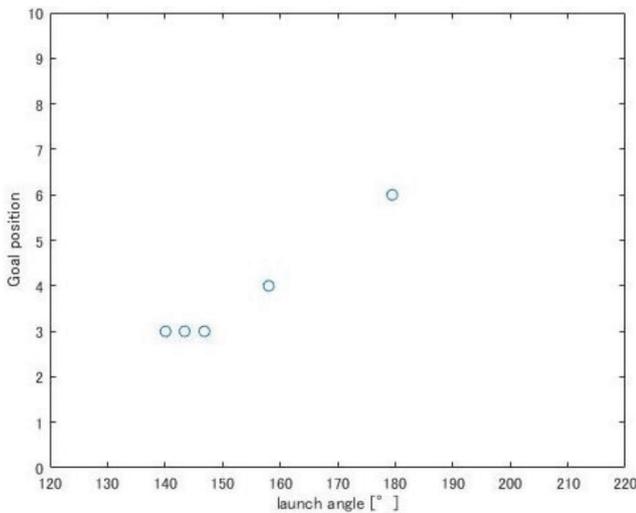


Figure 6 | The result of learning five times.

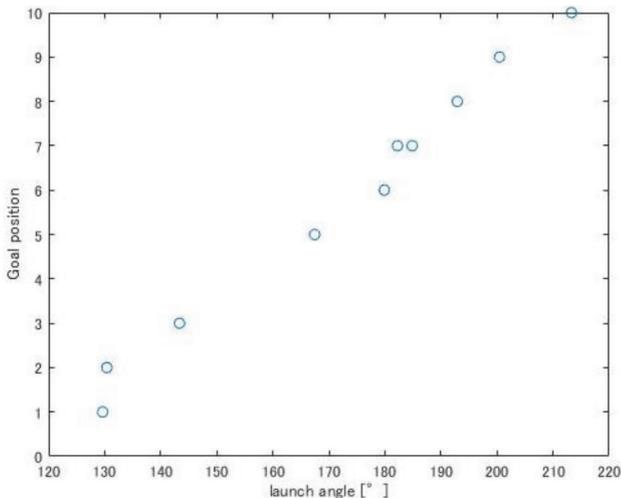


Figure 7 | The result of learning 10 times.

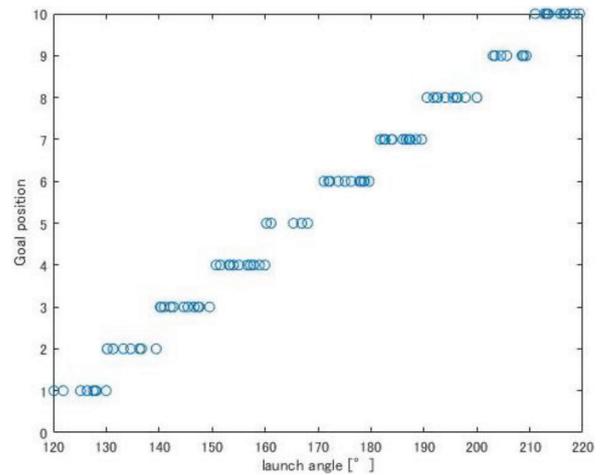


Figure 8 | The result of learning 100 times.

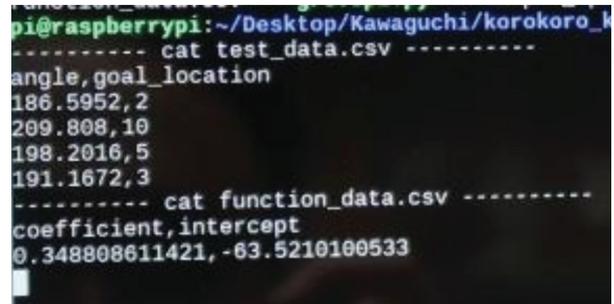


Figure 9 | Output results on Raspberry Pi.

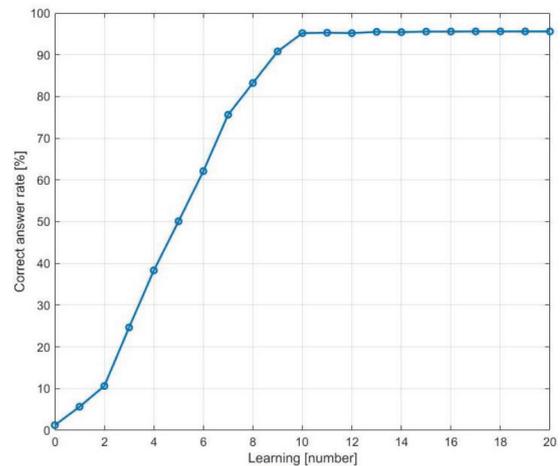


Figure 10 | Relationship between learning frequency and correct answer rate.

where,  $N$  is the number of experiments, and  $TP$  is the number of times that the keeper robot catches the ball.

From these results, we can see that the correct answer rate increases as the number of learning times increases. Based on these results, the keeper robot, as in the beginning no learning was performed, was unable to catch the ball, but after performing learning, inappropriate actions gradually disappear, and eventually it becomes able to catch the ball.

## 5. CONCLUSION

In this paper, we proposed a method of estimating the target value for a robot controlling the trajectory of a ball using machine learning. The result of this is that, as the number of learning times increase, the accuracy can be increased, and we were able to confirm the effectiveness of the proposed method through experimentation. Moving forward, we plan to look into the application of the method in a separate device to create a versatile system. Additionally, we plan to provide support for models other than the linear regression model, such as reinforced learning and deep learning etc.

## CONFLICTS OF INTEREST

The author declares no conflicts of interest.

## ACKNOWLEDGMENTS

This work was supported by enPiT. Thank you for the Smart SE: Smart Systems and Services innovative professional Education program. We asked students from Tokyo Gakugei University to cooperate in the development of this device.

## REFERENCES

[1] R.P. Herrero, J.P. Fentanes, M. Hanheide, Getting to know your robot customers: automated analysis of user identity and

- demographics for robots in the wild, *IEEE Robot. Autom. Lett.* 3 (2018), 3733–3740.
- [2] E. Dean-Leon, K. Ramirez-Amaro, F. Bergner, I. Dianov, G. Cheng, Integration of robotic technologies for rapidly deployable robots, *IEEE Trans. Ind. Inform.* 14 (2018), 1691–1700.
- [3] K.H. Yeh, A secure IoT-based healthcare system with body sensor networks, *IEEE Access* 4 (2016), 10288–10299.
- [4] K. Ashton, That ‘internet of things’ thing, *RFID J.* 22 (2009), 97–114.
- [5] B. Vallade, A. David, T. Nakashima, Three layers framework concept for adjustable artificial intelligence, *J. Adv. Comput. Intell. Intell. Inform.* 19 (2015), 867–879.
- [6] A. Nareyek, AI in computer games, *Queue* 1 (2004), 58–65.
- [7] S.K. Chalup, C.L. Murch, M.J. Quinlan, Machine learning with AIBO robots in the four-legged league of RoboCup, *IEEE Trans. Syst. Man Cybernet. C* 37 (2007), 297–310.
- [8] F. Çakmak, E. Uslu, M. Fatih Amasyali, S. Yavuz, Hazmat charts detection on search and rescue league of RoboCup competitions, 2017 25th Signal Processing and Communications Applications Conference (SIU), IEEE, Antalya, Turkey 2017, pp. 1–4.
- [9] S.D. Ramchurn, A. Farinelli, K.S. Macarthur, N.R. Jennings, Decentralized coordination in RoboCup rescue, *Comput. J.* 53 (2010), 1447–1461.
- [10] Y. Chen, Y. Li, Macro and micro reinforcement learning for playing nine-ball pool, 2019 IEEE Conference on Games (CoG), IEEE, London, UK, 2019.
- [11] S. Lee, H. Lim, Y. Kim, H. Myung, Effective strategies for improving mobility efficiency and keeping numerical superiority in AI worldcup, 2019 16th International Conference on Ubiquitous Robots (UR), IEEE, Jeju, South Korea, 2019.

## AUTHOR INTRODUCTION

### Dr. Shinichi Imai



He graduated doctor course at department of engineering in Hiroshima University. He works at department of education in Tokyo Gakugei University. His research area is about control system design, educational engineering.