Ichiro TeenSize Team Description Paper 2019

Muhtadin[1], Sulaiman Ali[1], Muhammad Reza Arrazi[1], Tommy Pratama[1], Dhany Satrio Wicaksono[1], Ahmad Hernando Pradanatta Putra[1], I Made Pande Ari[1], Alfi Maulana[1], Oktaviansyah Purwo Bramastyo[1], Syifaul Qolby Asshakina[1], Dzulfikar Ats Tsauri[1], Arsy Huda Fathaniard[1], Sinta Devi Listianah[1], Muhammad Attamimi[1], Muhammad Arifin[2]

[1] Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia
[2] Indonesian Institute of Sciences (LIPI), Indonesia
muhtadin@ee.its.ac.id, sulaimanali281@gmail.com

Abstract. This paper describes the Ichiro RoboCup Humanoid League team Ichiro TeenSize from the Institut Teknologi Sepuluh Nopember Surabaya, Indonesia. We made this paper to fulfill the prerequisites for participation in Robocup 2019 which will be held in Sydney, Australia. This paper presents a brief overview of our existing software and hardware of our robots that will be used in the competition. Especially in terms of developing mechanics, electronics, cooperation between robots, as well as improving the vision system for our robots.

1 Introduction

The ultimate goal of the RoboCup Humanoid competition is to realize a robot soccer team against the human soccer team in accordance with FIFA rules, encouraging many researchers to study the development of humanoid robots in various aspects, such as the development of reliable mechanical design, the ability of the robot to interpret the environments, visual perception in a complex environment, as well as the ability to localize and coordination between the robots for scoring as many goals as possible. The RoboCup's ultimate goal attracts us as researchers to participate in the development of humanoid robots, as well as attracting our interest in joining the Humanoid Robot competition both on a National and International levels.

Team Ichiro specifically develops research in the field of Humanoid Robotics. Members of the team Ichiro are students in undergraduate and diploma programs from the Institut Teknologi Sepuluh Nopember Surabaya, Indonesia. We have participated in various competitions of Humanoid robots at the National level starting in 2014. We began participating in the International level in 2016. In that year we received 10 awards with two world records at FIRA Robowoldcup 2016 in Beijing China. In the same year, we qualified for the RoboCup competition in Leipzig Germany but failed to leave due to lack of funds to dispatch the team to Germany. In 2017, we successfully participated in the RoboCup Humanoid League competition in the KidSize category held in Nagoya

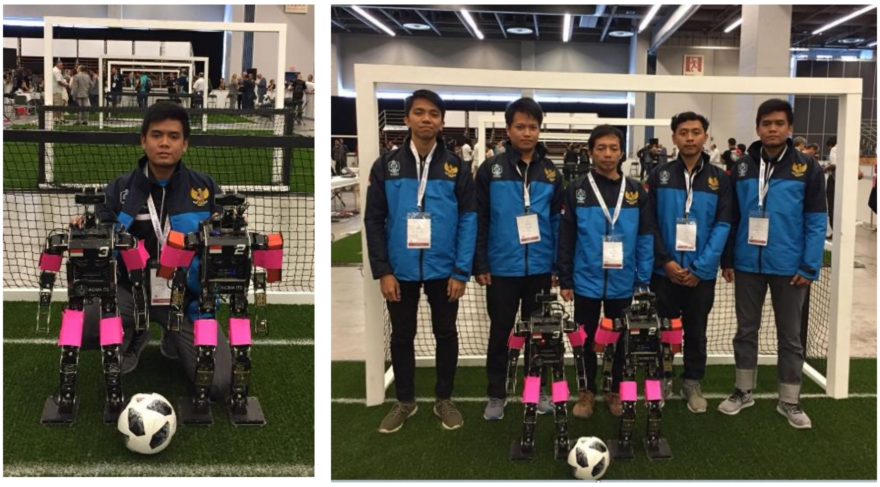

Fig. 1. Left: Ichiro Robot TeenSize. **Right:** Robot Team Member at Robocup 2018.

Japan. Our participation in the competition opened many insights for us about the development of humanoid robots. In 2018, we had problems with limited funds to dispatch large numbers of teams to Canada, therefore, we took part in the RoboCup Humanoid League competition in the TeenSize category (see Fig. 1) which can be done by only two robots with a small number of teams. The amazing results had been achieved from the RoboCup competition in Canada. We won several awards: First place in the TeenSize category, Runner-up at the TeenSize Technical Challenge, Runner-up at the Drop-in Games, finally, we won the third place in Best Humanoid Robot Soccer.

In this paper, we will give an overview of the Ichiro Robots hardware and software system and its current state, and the development of robots that we are currently doing to participate in the competition in RoboCup 2019 in Sydney, Australia.

2 Mechanical Hardware Overview

Ichiro TeenSize was made based on a modification from Nimbro-OP [1]. We still use these robots because of their robustness and stability. Thanks to our stability control design so that the robot able to perform a long shot.

This robot uses one Logitech camera C922 and two LiPo 4S batteries with 300 grams weight. There are 20 degrees of freedom using Dynamixel Servo. Twelve Dynamixel MX 106 for the robot's feet, six Dynamixel MX 64 for the robot's arms and two Dynamixel MX 28 for the head. Ichiro Teen Size V.1 has 0.85 m of height. This robot is made by cutting and bending with 3 mm thick type 5 Aluminum material. To support the balance of the robot, we used four load cells with a capacity of 10 kg for each leg. We made the loadcell system based on the Rhoban Football Club Team's foot sensor system [2].

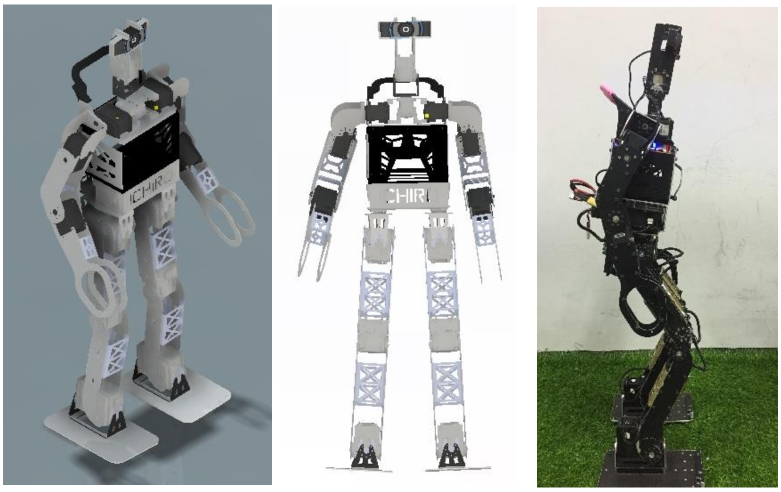

Fig. 2. Left and **Centre:** CAD Design of Ichiro TeenSize. **Right:** Ichiro TeenSize.

At the meaning time, we have succeeded in improving our robot mechanics (see Fig. 2). Innovations were made because of the existence of robots that have low running speeds. This robot has a height of 0.84 m and weighs 7.2 kg. This robot has an improvement in the body and upper arm. Enhancement is done by reducing weight. This improvement will affect the position of the center of gravity (COG). By lowering the COG, robots are expected to run faster. The arm is changed to protect the servo when the robot falls. Ichiro TeenSize uses aluminum type 5 with 3 mm thickness for legs and type 6 aluminum with 2 mm thickness for other body parts.

3 Electrical Hardware Overview

Ichiro Robot TeenSize uses an Intel NUC mini PC as its main controller. For the visual sensor, we are using a Logitech C922 and C920 camera that plugged on the USB port on Intel NUC.

Since 2017 we have not used a compass because of the regulations that have been applied. For the orientation sensor, we have used MPU-6050 where this sensor is quite accurate with a 16-bit analog to digital converse internal hardware facilities for each channel. This sensor combines the 3-axis gyroscope and 3-axis accelerometer on the same chip. To be able to interact with MPU-6050, a microcontroller such as Arduino is needed as an interface for i2c-bus. In our robot, we used Arduino nano as an access to this sensor.

For joints movements in our robots, it is driven by a servo motor. There are 20 joints on the robot and we use a combination of Dynamixel MX-28 servo, Dynamixel MX-64 servo, and Dynamixel MX-106 servo. There are six Dynamixel MX-106 servo motors on each leg, 3 Dynamixel MX-64 servo motors are used in each arm and there is two Dynamixel MX-28 servo motor for the neck. To interact with a servo motor, we have used a sub-controller in the form of a cm-740.

This robot requires supply power with a voltage range of around 12 volts to 19 volts. We use a 4-cell LiPo battery, 3300 mAh. Supply power is intended for Intel NUC and cm-740, but for cm-740 we only use a supply power of 16 volts.

For the current version of our robot, the control system that we used is the same. We did not make any changes in electrical terms. We only minimized the board by combining the power switch board and MPU-6050 board into one to make it more compact and minimalist.

4 Vision

In 2018 we used a very different method from the previous year. Our previous method used ball detection by filtering it according to color, shape, and size. However, due to the significant changes in the RoboCup regulation, specifically using the dominant white ball, white goal post, and synthetic grass, the previous method was no longer efficient to be used.

In RoboCup 2018, we make a more independent light detection method, high accuracy, low computation and do not require a lot of tuning. For ball detection, we use a Local Binary Pattern (LBP), which is a texture descriptor popularized by Ojala et al. [3]. Unlike the Haralick texture feature that calculates global texture representations based on the Gray Level Co-occurrence Matrix, LBP calculates local texture representations. This local representation is built by comparing each pixel with the surrounding pixels. The result of ball detection is shown in Fig. 3.

For the classification process, we use the cascade classifier [4][5]. Cascading classifiers are trained with hundreds of "positive" and "negative" images of the same size. This method is very suitable to run on a low-power CPU because it has a fast processing speed.

To reduce noise outside the field, we first segmented the color of the field based on the color. From the contour detected, the contour that had the largest area was selected

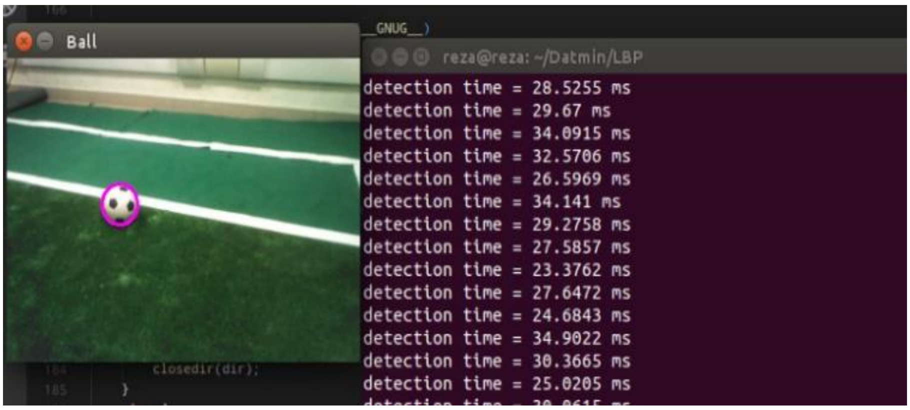

Fig. 3. LBP-based ball detection

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

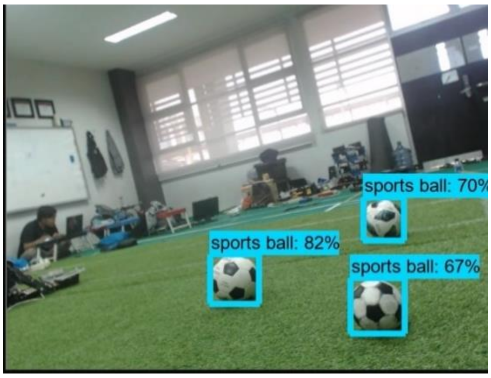

Fig. 4. Left: Network layer of MobileNet V1**. Right:** CNN-based Ball Detection

and then Convex Hull was performed. After that, we classify it using the LBP feature on the object inside convex.

With this method, satisfying results have been achieved. The robot could detect balls in a maximum distance of 400 cm with a detection time up to 19 ms.

For 2019 RoboCup Humanoid League Competition, we also try another method to detect many objects in the soccer field. We use the architecture of Mobilenet v1 [6] and use a Single Shot multi-box Detector (SSD) for object localization. Fig. 4 shows the implementation of Mobilenet v1 and the implementation result in Ichiro robot.

5 Robot Behavior and Strategy

5.1 Robot Monitoring System

Behavior and strategy are determined in many states. The transfer of state robots to the next state is determined by various information, such as the number of teammates

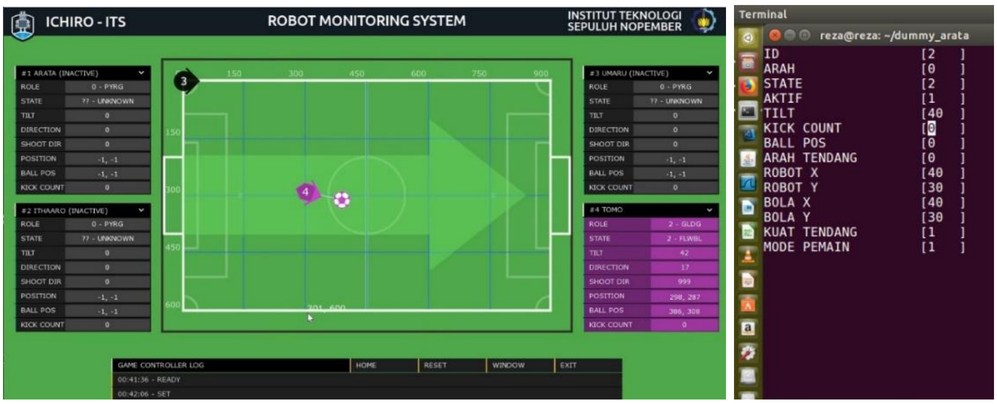

Fig. 5. Software for Monitoring

on the field, game status from the game controller, and the individual teammates robot's state. Because we have difficulty monitoring many robotic states based on information received by robots, we create a system to monitor the robot state. GUI of Ichiro's robot monitoring system shown in Fig. 6. We also make a "dummy" robot to send fake information when debugging robot behavior. GUI of "dummy" robot shown on Fig. 5.

5.2 Walking Engine and Localization Method

We implemented the sinusoidal trajectory to our robot's walk engine. This movement doesn't use dynamic modeling of the robot so that this walk engine is open loop and does not use ZMP criterion as described in [7]. Based on the given points of trajectory, all the joints are computed by the inverse kinematics of the legs of the robot. Due to the imperfection of actual dynamics of the robot, we have to tune some parameters in walk engine manually with trial and error. We also implemented the PD control strategy on both arms and hips of the robot to maintain its pitch at the desired angle to prevent the robot from falling.

Sensors data and localization module information are needed to design more complex robot behavior. First, we need to use motion capture camera to capture robot's displacement based on given gait command for implementing local localization of the robot. We use Optitrack® Trio. The data is collected by putting the markers on both feet, then we track those markers using the motion capture system. The generated data by motion capture is then processed using machine learning with robot's forward kinematic to get actual each displacement of leg while the robot is walking (see Fig. 6). Based on these motion capture data, we can predict the robot step model based on the gait parameters we use. This method has been discussed in more detail in [8].

Based on the predictions of the current step model that we have obtained, by fusing this data and orientation data from robot's sensor, we are able to get a quite accurate location of the robot which is used as information of localization module. This method, however, needs the initial location of the robot on the field which is obtained by observing both locations of goal posts at the beginning of the game or whether the robot starts to enter the field. The distance of both goal posts then compared relative to the robot to obtain the own fields and location of the robot as described in [9].

When the robot starts entering the field, robot use some possible initial, this method has also been carried out by the Nimbro team [8]. According to the fact that robots always entering their own area when the game starts, our robot will calculate the distance between the robot and the goalposts using the trilateration method as discussed in [9]. Our robot will choose several initial positions that has been defined based on the estimated distance of the robot with the own goal post and/or opponent's goal post.

The error of the robot position often occurs when the robot plays for a long time (about 2 minutes). However, we can solve this problem by re-estimating the initial position when the game state is initial, which is usually happen when a goal or a drop ball occurs. Re-estimating the initial position also done when the robot coming to the games after they had been picked-up or service.

5.3 Robot Behavior and Strategy

The finite state machine is used to design the robot behavior which its transitions are different based on current game state, teammates states or location and orientation information from localization module. The communication between the robots is implemented using UDP communication which contains information of robot's states. Then in general, the robots are considered as two roles as defender and striker robots. The defender exists if the striker is active on the field. To reduce power consumption and prevent a condition where our field is empty of player, the defender will stay on his determined position and approach the ball when the ball is around him. The defender robot is able to give the information of the location of the visible ball to striker robot if striker robot has not found the ball yet. To improve the effectiveness of ball exploration on the field, the striker robot will search the ball on the given point coordinates, which are own field, the center of the field and opponent's field. Striker robot is also able to aim the ball kick by using estimated position and orientation of the robot.

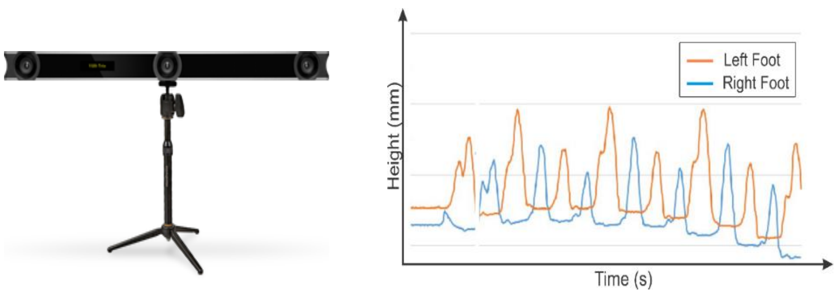

Fig. 6. Left: Optitrack Trio Motion Capture**. Right:** Generated data by Optitrack Trio Motion Capture

References

1. Max Schwarz, Julio Pastrana, Philipp Allgeuer, Michael Schreiber, Sebastian Schueller, Marcell Missura and Sven Behnke: Humanoid TeenSize Open Platform NimbRo-OP In Proceedings of 17th RoboCup International Symposium, Eindhoven, Netherlands (2013).
2. Julien Allali, Louis Deguillaume, R´emi Fabre, Loic Gondry, Ludovic Hofer, Olivier Ly, Steve N'Guyen, Gr´egoire Passault, Antoine Pirrone, and Quentin Rouxel. Rhoban Football Club: RoboCup Humanoid Kid-Size 2016 Champion Team Paper, pages 491–502. (2017).
3. T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, (2002).
4. Paul Viola and Michael J. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. In IEEE Computer Vision and Pattern Recognition CVPR, (2001).
5. Rainer Lienhart and Jochen Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. (2002).
6. A. Howard. Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861, (2017).
7. Sven Behnke. Online trajectory generation for omnidirectional biped walking. In 2006 IEEE International Conference on Robotics and Automation (ICRA 2006), pages 1597–1603. IEEE, (2006).
8. Ficht, Grzegorz & Pavlichenko, Dmytro & Allgeuer, Philipp & Farazi, Hafez & Rodriguez, Diego & Brandenburger, André & Kürsch, Johannes & Schreiber, Michael & Behnke, Sven. Grown-up NimbRo Robots Winning RoboCup 2017 Humanoid AdultSize Soccer Competitions. RoboCup 2017: Robot World Cup XXI, pages 448-460, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg (2017).
9. Schmitz A., Missura M., Behnke S. "Learning Footstep Prediction from Motion Capture". In: Ruiz-del-Solar J., Chown E., Plöger P.G. (eds) RoboCup 2010: Robot Soccer World Cup XIV. RoboCup 2010. Lecture Notes in Computer Science, vol 6556. Springer, Berlin, Heidelberg (2011)
10. Muhammad Arifin, Muhammad Ardi Pradana, Uti Solichah, Satria Hafizhuddin, Rizal Hafidhilllisan, Dhany Satrio Wicaksono, Muhammad Reza Ar Razi, Rudy Dikairono Muhtadin, "ICHIRO Kid-Size 2016 Team Description Paper". (2016).
11. J. M. Ibarra Zannatha, R. Cisneros Limón, A. D. Gómez Sánchez, E. Hernández Castillo, L. E. Figueroa Medina and F. J. K. Lara Leyva, "Monocular visual self-localization for humanoid soccer robots," CONIELECOMP 2011, 21st International Conference on Electrical Communications and Computers, San Andres Cholula, pp. 100-107 (2011).