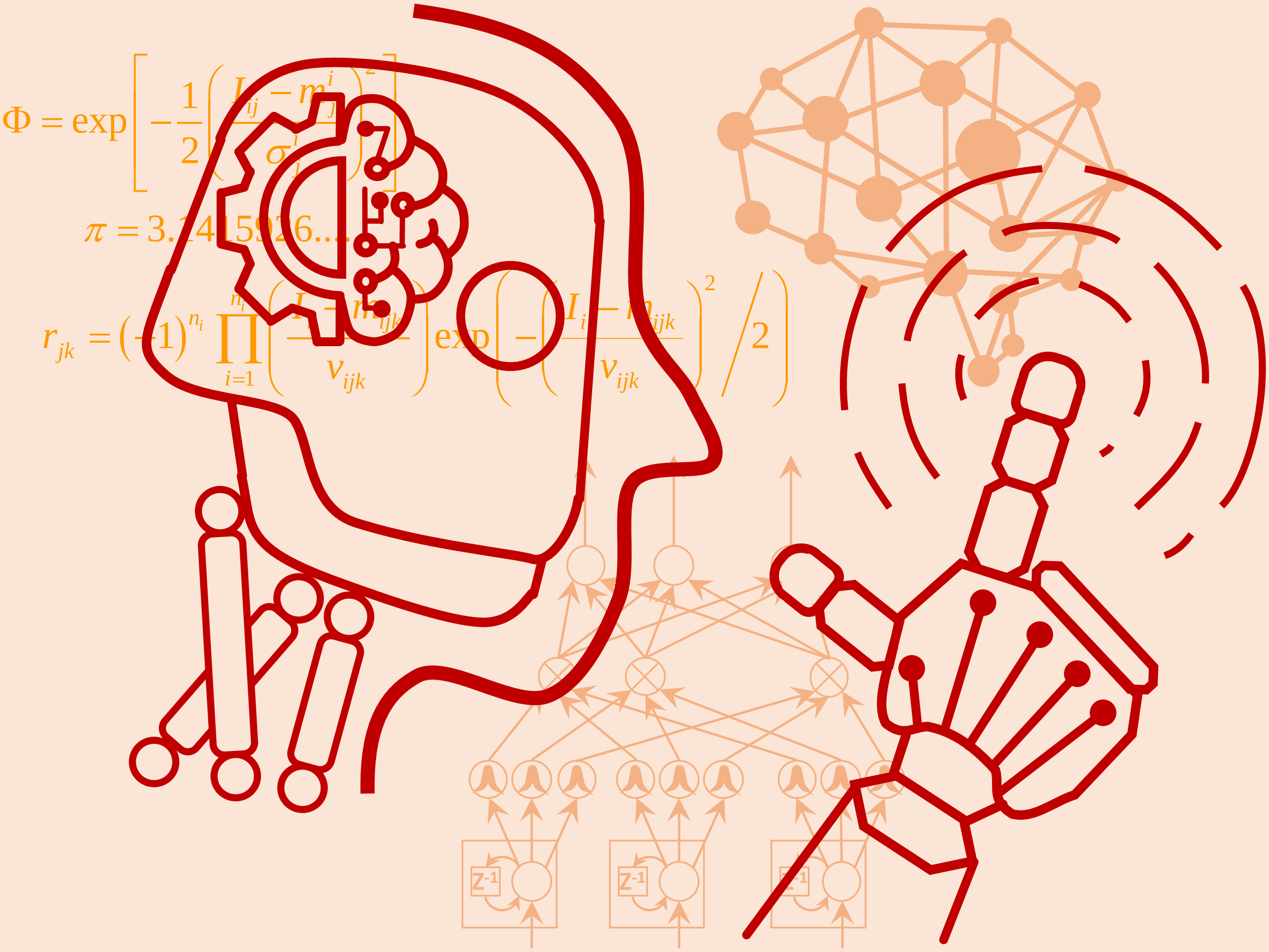
ISSN 2616-8170

RST
Robotics Society of Taiwan

iRobotics

VOLUME 2
NUMBER 3
SEPTEMBER 2019


$\Phi = \exp\left[-\frac{1}{2}\left(\frac{I_{ij} - m_j^i}{\sigma_j^i}\right)^2\right]$
$\pi = 3.1415926...$
$r_{jk} = (-1)^{n_i} \prod_{i=1}^{n_i} \left(\frac{I_i - m_{ijk}}{v_{ijk}}\right) \exp\left(-\left(\frac{I_i - m_{ijk}}{v_{ijk}}\right)^2 \Big/ 2\right)$
Z-1
Z-1
Z-1

PUBLISHED BY THE ROBOTICS SOCIETY OF TAIWAN

*i*Robotics

EDITORIAL BOARD

Editor-in-Chief

Editors

PUBLISHER

Robotics Society of TAIWAN (RST)
Society President: Ching-Chih Tsai

Department of Electrical Engineering, National Chung Hsing University
Taichung, Taiwan

Tel: +886-4-2285-1549#601
URL: http://www.rst.org.tw

The ***i*Robotics** is published quarterly each year by the Robotics Society of Taiwan (RST). Institutional rate: US$140 annually; individual annual subscription rate: US$50 for nonmembers, US$25 for members (including postage). Note that another US$100 is needed if the express is required.

EtherCAT-based Impedance Control and Force Compliance Teaching of a 6-DOF Industrial Robotic Manipulator

Ching-Chih Tsai, *Fellow*, *IEEE*, Chun-Chieh Chan, Chun-An Lin, and Feng-Chun Tai

***Abstract*—This paper presents an EtherCAT-based impedance control method and force compliance teaching for a 6-degrees-of-freedom (DOF) industrial robotic manipulator. After describing the system structure of this impedance controller using EtherCAT protocol, the forward kinematics of the manipulator is then derived with its Denavit-Hartenberg (DH) parameters. Based on the geometrical configurations of the robotic arm and forward kinematics model, an analytically inverse kinematics method is proposed to solve the joint angles of the manipulator moving from one pose to another. For impedance control, a proportional-integral (PI) impedance controller along with a 6-DOF force/torque sensor and six independent proportional–integral–derivative (PID) torque controllers are proposed to achieve impedance control with satisfying performance. For force compliance teaching, a torque generation method is presented using gravity and friction compensation, forward and inward dynamics of the manipulator. Simulation results are conducted to show the effectiveness and merits of the proposed inverse kinematics method, impedance controller and force compliance teaching. The applicability and practicability of the proposed method are also well exemplified by conducting experimental results on a 6-DoF industrial manipulator (RA605) from HIWIN technologies corp..**

***Index Terms*—impedance control, compliance control, compliance teaching, human-robot cooperation**

I. Introduction

To date, six DOF robotic manipulators have been widely developed not only for industrial automation, but also for service robots in our daily life. Many companies, such as Fanuc, Mitsubishi, Yaskawa, Kuka and ABB, have already developed six-DOF robotic manipulators with advanced functions including vision, force control, intelligent gripping and even human-robot collaboration. Such advanced 6-DOF robot manipulators can be programmable to perform complicated tasks for a wide range of applications. For automatic or semi-automatic manufacturing of 3C commercial products, many advanced 6-DOF robotic manipulators have been utilized for welding, digging, cutting, polishing, holding, carrying, processing, and assembling, in order not only to reduce labor costs but also to accomplish complex work possibly.

EtherCAT is a kind of industrial Ethernet protocol that has recently evolved into numerous solutions and standards for industrial automation and real-time control systems for both industrial and service robots [1-3]. EtherCAT retains the technical merits of Ethernet, such as low cost of infrastructure components, variable topologies allowing various distributed control schemes, many silicon vendors and, most importantly, much higher bandwidth and speed than traditional fieldbus solutions. Moreover, EtherCAT network is based on Ethernet, industrial field-bus protocol with 100Mbps transfer speed, and has short response time and short cycling time in the robot environment due to its functional principle processing on the fly. Furthermore, EtherCAT allows several hundreds or thousands of I/O devices inside one a network. Researchers showed the effectiveness of the EtherCAT network for human-robot cooperation and collaboration, and real-time control systems [1-3]. Inspired by [1-2], EtherCAT can be expected to play an important role in position and force control of 6-DOF robotic manipulators.

Historically, force control approaches can be categorized into three classes: hybrid position/force control, impedance control, and hybrid impedance control (HIC)[4-7].To achieve impedance control of robotic manipulator, researchers have proposed several control techniques which can be classified into two technical categories: position-based [8-11] and torque-based [12-15]. Generally speaking, both categories are based on dynamic model of robotic manipulator. In particular, the authors in [9] proposed a kinematic-based impedance control without considering the dynamic behavior of the robotic manipulator. This approach would be useful when the dynamic model of a robotic manipulator is difficult to establish. However, this type of impedance control requires the calculations of the forward and inverse kinematics (IK).

The forward kinematics problem is straightforward to derive the equations if all joint angles are known. The conversion of the position and orientation of a manipulator end-effector from Cartesian space to joint space is called as the IK problem. In general, the forward kinematics of a robotic manipulator can be easily derived if its DH parameters are determined, while its IK problem can be addressed using its physical structure and continuous movement in order to avoid multiple solution issues. The authors in [16] proposed an analytical IK method using field-programmable-gate-array (FPGA) devices. Other algorithms, such as Jacobian inversion, genetic programming, and biological computational approaches, have also been developed to solve IK problems [17-19]; they may be useful in some application domains.

Ching-Chih Tsai, Chien-Cheng Yu, Shih-Min Hsieh and Feng-Chun Tai are with the Department of Electrical Engineering, National Chung Hsing University, Taichung, 40227, Taiwan.

The authors deeply acknowledge finance support from the Ministry of Science and Technology (MOST), Taiwan, ROC, under contract MOST 107-2218-E-005 -005 -

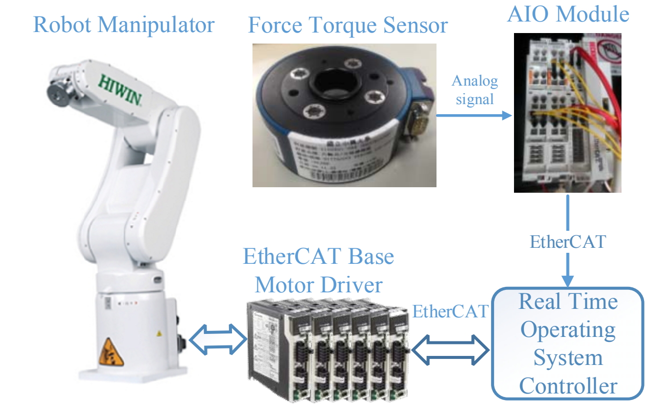


Figure 1. Physical system structure of the proposed impedance control system for the HIWIN 6-DOF robotic manipulator.

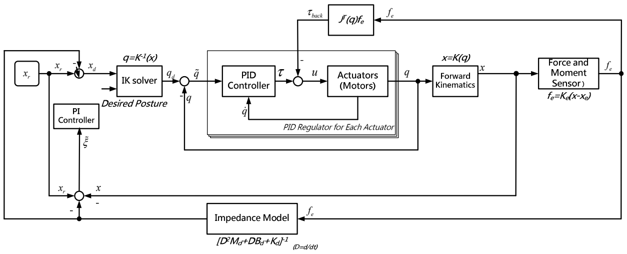


Figure 2. Block diagram of the impedance control system for the HIWIN 6-DOF robotic manipulator.

On the other hand, robot teaching can be extensively investigated by many approaches, including conventional teaching pedant, robotic language [20], learning by demonstration [21-24] and force-based teaching [25-30]. Force compliance teaching has become increasing important owing to its intuitive advantages in recent years [27,30-31]. From technical viewpoint, force-based teaching can be roughly divided into two groups: direct and indirect. Direct force compliance teaching refers to the use of force sensors to measure the external applied force and then find the corresponding torque values of all the joints, while indirect method uses indirect force sensors, such as current sensing devices, to estimate the exerting force and, similarly, obtain the joint torque values for movement of the manipulator. Generally speaking, the indirect method is much cheaper than the direct one due to inexpensive sensing or sensorless technologies. This direct approach may be useful when the force sensor of a robotic manipulator is available for other force-related applications [32].

The objectives of this paper are to develop a kinematic impedance control method and a force compliance teaching method for an EtherCAT-based 6-DOF robotic manipulator from HIWIN. The presented contents are written in three technical contributions. One is the development of a new IK algorithm to find all joint angles of the manipulator. The second contribution is design, numerical and experimental verification of the position-based impedance control method by utilizing the developed forward kinematics and IK algorithms, a PI impedance controller and a 6-DOF force/torque sensor together with six independent PID torque controllers. The other is the system integration of a force-based teaching approach with gravity and friction compensation, forward and inward dynamics for the manipulator.

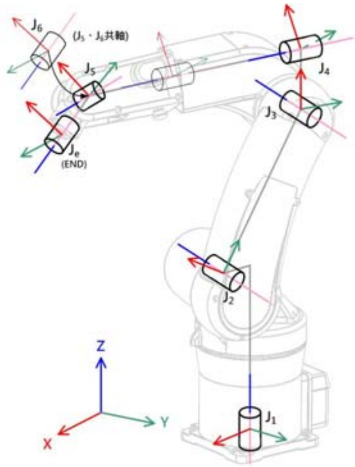


Figure 3. Overall structure of the HIWIN 6-DOF robotic manipulator.

TABLE 1.
DH PARAMETERS OF THE HIWIN 6-DOF ROBOTIC MANIPULATOR.

Joint *j*	α_j	d_j **(mm)**	a_j **(mm)**	$\theta_{j,min} < \theta_j < \theta_{j,max}$
J1	90	375	30	-165° < θ_1 < 165°
J2	0	0	340	-55° < θ_2 < 185°
J3	90	0	40	-55° < θ_3 < 185°
J4	-90	338	0	-190° < θ_4 < 190°
J5	90	0	0	-115° < θ_5 < 115°
J6	0	86	0	-360° < θ_6 < 360°

The rest of this paper is organized as follows. The impedance control system structure of the 6-DOF robotic manipulator is described in Section II, and the forward kinematics is introduced in Section III, and Section IV investigates the inverse kinematics and of this robot manipulator. Section V proposes and the impedance control method. In Section VI, force compliance teaching method is proposed using gravity compensation, friction compensation, forward dynamics, inward dynamics and the torque generation method of this robotic manipulator. Simulations are conducted in Section VII to show the performance on the proposed impedance control method and force compliance teaching method. Experimental results are presented in Section VIII. Section IX concludes this paper.

II. IMPEDANCE CONTROL ARCHITECTURE

This section is devoted to introducing the physical control system and impedance control architecture of the six degrees-of-freedom (DOFs) robotic manipulator from HIWIN. Figure 1 shows the overall physical structure of the HIWIN 6-DOF robotic manipulator using the well-known EtherCAT network. EtherCAT network with 100Mbps is used for real-time communication which enables to realize both position and impedance control. Via EtherCAT network, the measured force signals from the a six-DOF force and moment sensor are transmitted to the IPC-based controller with a real-time operation system, called RTX, and the generated impedance control commands are then sent to the EtherCAT-based motor control drives, thereby carrying out desired motion control tasks in real time. Both impedance and force control laws are realized by the control rate within the range of 500-1000μs, which is the main domain for most EtherCAT applications.

Figure 2 depicts the block diagram of the impedance control system for the HIWIN 6-DOF robotic manipulator, where the environment around the robot is modeled as a high stiffness spring expressed by $f_e = K_e\,(x - x_e)$ where K_e represents the stiffness of an object located at x_e. This linear relationship between force and deformation represents a simple robot-environment modeling. In Figure 2, $\tilde{\xi}$ represents the auxiliary impedance error which will be shown to be zero at the steady-state condition, and the IK solver is proposed in Section IV, in order to find the corresponding six joint angles. The joint angle tracking modules are done by six PID regulators which respectively generate the torque control commands for the motor drivers in the presence of the constraint forces from the robot-environment model.

III. Forward Kinematics

A. Denavit-Hartenberg (DH) Parameters

This section is devoted to describing the forward kinematics of the robot, as shown in Figure 3. The forward kinematics method is aimed to obtain the mathematical relationships between the given joint angles,the position and posture of the end-effector. It is usual that DH parameters are employed to obtain the forward kinematics equation of the manipulator. DH parameters are standard descriptions of the geometric positions/ pose of joints and links in the three-dimensional Cartesian coordinates. In doing so, The N links of the manipulator are numbered from 1 to N. Let θ_j, a_j, d_j and α_j be the four key parameters of link j and joint j individually; θ_j denotes the joint angle, a_j the link offset, d_j the link length, α_j represents the link twist. Table I lists all the DH parameters of the robotic manipulator.

In deriving the forward kinematics of the manipulator, the homogeneous transformation matrix is defined in the following equation based on the j^{th} link parameters of the manipulator.

$$\begin{aligned} {}^{n-1}T_n &= Trans_{zn-1}(d_n)\cdot Rot_{zn-1}(\theta_n) Trans_{xn}(a_n) Rot(\alpha_n) \\ &= \begin{bmatrix} \mathrm{c}\theta_n & -\mathrm{s}\theta_n \mathrm{c}\alpha_n & \mathrm{s}\theta_n \mathrm{s}\alpha_n & a_n \mathrm{c}\theta_n \\ \mathrm{s}\theta_n & \mathrm{c}\theta_n \mathrm{c}\alpha_n & -\mathrm{c}\theta_n \mathrm{s}\alpha_n & a_n \mathrm{s}\theta_n \\ 0 & \mathrm{s}\alpha_n & \mathrm{c}\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} = \left[\begin{array}{ccc|c} & R & & T \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \end{aligned} \quad (1)$$

where $\mathrm{s}\theta_n = \sin\theta_n$, $\mathrm{c}\theta_n = \cos\theta_n$.

B. Forward Kinematics

This subsection is dedicated to find the forward kinematics of the 6-DOF manipulator, which adopts the rotation R of the rigid body and uses the triple of angles (α, β, γ) to represent the rotation. Note that α, β, and γ are the Euler angles of the end effector in the world coordinates.

The transformation matrix of the manipulator can be expressed by the product of the following homogeneous transformation matrices.

$${}_0^6T = {}_0^1T \times {}_1^2T \times {}_2^3T \times {}_3^4T \times {}_4^5T \times {}_5^6T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Equation (2) defines the transformation matrix of the end-effector from its coordinate frame to the base coordinate frame. Note that n, o, x respectively represent the posture vectors, and p denotes the position of the end-effector. The position and posture of the manipulator is expressed by

$$\begin{aligned} \begin{bmatrix} x & y & z & \alpha & \beta & \gamma \end{bmatrix}^T &= \mathbf{h}(\theta_1,\theta_2,\theta_3,\theta_4,\theta_5,\theta_6,\theta_7) \\ &= \begin{bmatrix} h_1(\theta_1,\theta_2,\cdots,\theta_7) & \cdots & h_6(\theta_1,\theta_2,\cdots,\theta_7) \end{bmatrix}^T \end{aligned} \quad (3)$$

where

$$h_1 = p_x \quad h_2 = p_y \quad h_3 = p_z$$

$$h_4 = \tan^{-1}\left(-\frac{a_x}{a_y}\right) \quad h_5 = \tan^{-1}\left(\frac{\sqrt{a_x^2 + a_y^2}}{a_z}\right) \quad h_6 = \tan^{-1}\left(\frac{n_z}{o_z}\right)$$

Furthermore, x, y and z represent the position of the end effector with respect to the world coordinates.

The physical meaning of the triple of angles (α, β, γ) is interpreted by using the two frames: A and B. First, rotate the frame B about the z- axis of frame B by an angle α, then rotate about the x- axis by an angle β, and finally rotate about the z-axis angle by an angle γ, thereby gives the frame A . This yields a pure rotation R of the rigid body with the triple of angles (α, β, γ) between both frames.

$$\begin{aligned} R_{AB} = R_z(\alpha) R_x(\beta) R_z(\gamma) &= \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} \in SO_3 \\ &= \begin{bmatrix} c\gamma c\alpha - s\gamma s\alpha c\beta & -c\gamma s\alpha c\beta - c\alpha s\gamma & s\beta s\alpha \\ c\gamma s\alpha + s\gamma c\alpha s\beta & c\gamma c\alpha c\beta - s\alpha s\gamma & -s\beta c\alpha \\ s\beta s\gamma & c\gamma s\beta & c\beta \end{bmatrix} \end{aligned} \quad (4)$$

IV. Inverse Kinematics

This section is aimed to solve for the inverse kinematics problem of the 6-DOF manipulator. Given the position and the Euler posture angles, the inverse kinematic problem is to solve for a set of six joint angles using (5) which is employed to find the end rotation matrix R, and (1) which is utilized to deduce ${}_0^6T$. Once the homogeneous matrix ${}_0^6T$ has been given, one can solve for the angles of the joints in the following six steps.

Step 1: In Figure 3, it is obvious that the rotations of the links 4, 5 and 6 do not change the position of the fifth joint. This implies the following equation:

$$\begin{aligned} {}_0^6T({}_5^6T)^{-1}({}_4^5T)^{-1} &= {}_0^1T \times {}_1^2T \times {}_2^3T \times {}_3^4T \\ &= \begin{bmatrix} r_{11} & r_{12} & r_{13} & \mathrm{a1{*}c1 + d4{*}c1{*}s23 + a2{*}c1{*}c2 + a3{*}c1{*}c23} \\ r_{21} & r_{22} & r_{23} & \mathrm{a1{*}s1 + d4{*}s1{*}s23 + a2{*}c2{*}s1 + a3{*}s1{*}c23} \\ r_{31} & r_{32} & r_{33} & \mathrm{d1 + a2{*}s2 - d4{*}c23 + a3{*}s23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5) \\ &= \begin{bmatrix} r_{11} & r_{12} & r_{13} & \mathrm{t_{14} - d6{*}t_{13}} \\ r_{21} & r_{22} & r_{23} & \mathrm{t_{24} - d6{*}t_{23}} \\ r_{31} & r_{32} & r_{33} & \mathrm{t_{34} - d6{*}t_{33}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

From the (1,4), (2,4) and (3,4)entries in (5), it follows that

$$\mathrm{c1(a1+a2c2+a3c23+d4s23)=J5_x = t_{14} - d6{*}t_{13}} \quad (6)$$

$$\mathrm{s1(a1+a2c2+a3c23+d4s23)=J5_y = t_{24} - d6{*}t_{23}} \quad (7)$$

$$\mathrm{d1 + a2s2 - d4c23 + a3s23 =J5_z =t_{34} - d6{*}t_{33}} \quad (8)$$

Dividing (7) by (8), one finds that the first joint angle is

$$\theta_1 = \tan^{-1}\left(\frac{\mathrm{t}_{24}\text{ - d6*}\mathrm{t}_{23}}{\mathrm{t}_{14}\text{ - d6*}\mathrm{t}_{13}}\right) \quad or \quad \tan^{-1}\left(\frac{\mathrm{t}_{24}\text{ - d6*}\mathrm{t}_{23}}{\mathrm{t}_{14}\text{ - d6*}\mathrm{t}_{13}}\right) + \pi \quad (9)$$

Worthy of mention is that if $\mathrm{J5}_x$=$\mathrm{J5}_y$=0, then $a1$+$a2c2$+$a3c23$+$d4s23$ must be equal to 0 and this results in a singular point, thereby providing infinitely many solutions.

Step 2: Summing the squares of (6) and (8) obtains

$$\begin{aligned} & a2^2 + a3^2 + d4^2 \\ & +2a3d4(c2s23 - s2c23) \\ & +2a2a3(c2c23 - s2s23) \end{aligned} = \left(\frac{\mathrm{J5}_\mathrm{x}}{c1} - a1\right)^2 + \left(\mathrm{J5}_\mathrm{z} - d1\right)^2 \quad (10)$$

$$\begin{aligned} 2a2a3c3 + 2a3d4s3 = & \left(\left(\mathrm{J5}_\mathrm{x}/c1\right) - a1\right)^2 + \left(\mathrm{J5}_\mathrm{z} - d1\right)^2 \\ & - a2^2 - a3^2 - d4^2 \end{aligned} \quad (11)$$

which leads to

$$\cos\left(\theta_3 - \tan^{-1}\left(\frac{d4}{a3}\right)\right) = \frac{\left(\left(\mathrm{J5}_\mathrm{x}/c1\right) - a1\right)^2 + \left(\mathrm{J5}_\mathrm{z} - d1\right)^2 - a2^2 - a3^2 - d4^2}{\sqrt{\left(2a2a3\right)^2 + \left(2a3d4\right)^2}} \quad (12)$$

From (12), one solve for the angle of joint 3 as

$$\theta_3 = \pm\cos^{-1}\left(\frac{\begin{bmatrix} \left(\mathrm{J5}_\mathrm{x}/c1 - a1\right)^2 + \left(\mathrm{J5}_\mathrm{z} - d1\right)^2 \\ -a2^2 - a3^2 - d4^2 \end{bmatrix}}{2a3\sqrt{\left(a2\right)^2 + \left(d4\right)^2}}\right) + \tan^{-1}\left(\frac{d4}{a3}\right) \quad (13)$$

Step 3: Since both θ_1 and θ_3 has been obtained, θ_2 can be found by summing the squares of (6) and (8) as:

$$\begin{aligned} & \mathrm{J5}_\mathrm{z} - d1 = \left(a3s3 - d4c3\right)c2 \\ & +\left(a2 + a3c3 + d4s3\right)\left(\frac{\left(\mathrm{J5}_\mathrm{x}/c1 - a1\right) - \left(a2 + a3c3 + d4s3\right)c2}{d4c3 - a3s3}\right) \end{aligned} \quad (14)$$

which leads to

$$\begin{aligned} & -\left(a3s3 - d4c3\right)^2 c2 - \left(a2 + a3c3 + d4s3\right)^2 c2 \\ & \quad + \left(\left(\mathrm{J5}_\mathrm{x}/c1\right) - a1\right)\left(a2 + a3c3 + d4s3\right) \\ & = \left(\mathrm{J5}_\mathrm{z} - d1\right)\left(d4c3 - a3s3\right) \end{aligned} \quad (15)$$

and then

$$c2 = \frac{\begin{bmatrix} \left(\left(\mathrm{J5}_\mathrm{x}/c1\right) - a1\right)\left(a2 + a3c3 + d4s3\right) \\ -\left(\mathrm{J5}_\mathrm{z} - d1\right)\left(d4c3 - a3s3\right) \end{bmatrix}}{\left(a2 + a3c3 + d4s3\right)^2 + \left(a3s3 - d4c3\right)^2} \quad (16)$$

Therefore, θ_2 is computed as follows:

$$\theta_2 = \cos^{-1}\left(\frac{\begin{bmatrix} \left(\frac{\mathrm{t}_{14} - \mathrm{d6}\times\mathrm{t}_{13}}{c1} - a1\right)\left(a2 + a3c3 + d4s3\right) \\ -\left(\mathrm{t}_{34} - \mathrm{d6}\times\mathrm{t}_{33} - d1\right)\left(d4c3 - a3s3\right) \end{bmatrix}}{\left(a2 + a3c3 + d4s3\right)^2 + \left(a3s3 - d4c3\right)^2}\right) \quad (17)$$

Step 4: With the three found joint angles, θ_1**,** θ_2 and θ_3 all elements of matrices ${}^{1}_{0}T^{-1}\,{}^{2}_{1}T^{-1}\,{}^{3}_{2}T^{-1}$ are known and they can be used to solve for other remaining angles. Thus, it follows that

$${}^{3}_{2}T^{-1}\,{}^{2}_{1}T^{-1}\,{}^{1}_{0}T^{-1}\,{}^{6}_{0}T = {}^{4}_{3}T\,{}^{5}_{4}T\,{}^{6}_{5}T \quad (18)$$

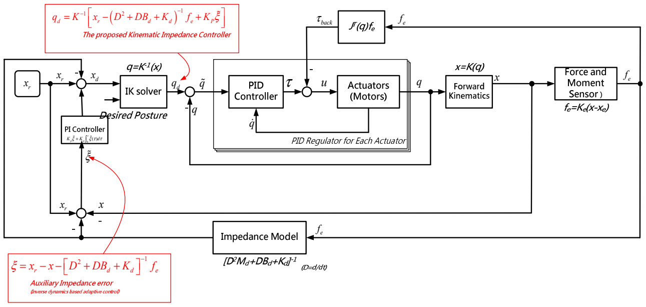

$q_d = K^{-1}\left[x_r - \left(D^2 + DB_d + K_d\right)^{-1} f_e + K_P\tilde{\xi}\right]$
The proposed Kinematic Impedance Controller
τ_{back}
$J^T(q)f_e$
f_e
x_r
x_r
x_d
q=K⁻¹(x)
IK solver
q_d
$\tilde{q}$
q
PID Controller
τ
u
Actuators (Motors)
q
$\dot{q}$
PID Regulator for Each Actuator
x=K(q)
Forward Kinematics
x
Force and Moment Sensor)
f_e
$f_e=K_e(x-x_e)$
Desired Posture
PI Controller
$K_P\tilde{\xi} + K_I\int_0^t \tilde{\xi}(\tau)d\tau$
$\tilde{\xi}$
x_r
x
Impedance Model
f_e
$[D^2M_d+DB_d+K_d]^{-1}$
(D=d/dt)
$\xi = x_r - x - \left[D^2 + DB_d + K_d\right]^{-1} f_e$
Auxiliary Impedance error
(Inverse dynamics based adaptive control)

Figure 4. System configuration of the proposed PI kinematic impedance control system for the HIWIN 6-DOF robotic manipulator.

which results in

$$\begin{bmatrix} \mathrm{c4c5c6 - s4s6} & \mathrm{-c6s4 - c4c5s6} & \mathrm{c4s5} & \mathrm{c4{*}d6{*}s5} \\ \mathrm{c4s6 + c5c6s4} & \mathrm{c4c6 - c5s4s6} & \mathrm{s4s5} & \mathrm{d6{*}s4{*}s5} \\ \mathrm{-c6s5} & \mathrm{s5s6} & \mathrm{c5} & \mathrm{d4 + c5{*}d6} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$= \begin{bmatrix} \times & \times & \begin{pmatrix} a_z\mathrm{s23} + a_y\mathrm{s1c23} \\ +\mathrm{c1}a_x\mathrm{c23} \end{pmatrix} & 0 \\ \times & \times & \left(a_x\mathrm{s1} - \mathrm{c1}a_y\right) & 0 \\ \begin{pmatrix} n_y\mathrm{s1s23} - n_z\mathrm{c23} \\ +\mathrm{c1}n_x\mathrm{s23} \end{pmatrix} & \begin{pmatrix} o_y\mathrm{s1s23} - o_z\mathrm{c23} \\ +\mathrm{c1}o_x\mathrm{s23} \end{pmatrix} & \begin{pmatrix} a_y\mathrm{s1s23} - a_z\mathrm{c23} \\ +\mathrm{c1}a_x\mathrm{s23} \end{pmatrix} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

From the element (3, 3) in (19), the joint angle θ_5 is computed by

$$\theta_5 = \pm\cos^{-1}\left(a_y s1s23 - a_z c23 + c1a_x s23\right) \quad (20)$$

Step 5: The joint angle θ_4 is solved from both elements (1, 3) and (2, 3) in (19) by

$$\theta_4 = \tan^{-1}\left(\frac{\left(a_x\mathrm{s1}\text{ - }\mathrm{c1}a_y\right)/s5}{\left(a_z\mathrm{s23} + a_y\mathrm{s1c23} + \mathrm{c1}a_x\mathrm{c23}\right)/s5}\right) \quad (21)$$

Step 6: Finally, the joint angle θ_6 is obtained from both elements (3, 1) and (3, 2) in (19)

$$\theta_6 = \tan^{-1}\left(\frac{\left(o_y\mathrm{s1s23} - o_z\mathrm{c23} + \mathrm{c1}o_x\mathrm{s23}\right)/s5}{-\left(n_y\mathrm{s1s23} - n_z\mathrm{c23} + \mathrm{c1}n_x\mathrm{s23}\right)/s5}\right) \quad (22)$$

V. PI KINEMATICS IMPEDANCE CONTROL

A. *PI Kinematic Impedance Control*

This section is devoted to developing the kinematic impedance control in presence of innegligible interaction force due to the constraints within its environment. Similar to [9], the proposed PI kinematic impedance control is aimed to control not only the dynamic behavior of the manipulator, but also its position or velocity. Figure 4 depicts the more detailed system configuration of the proposed PI kinematic impedance control system, which intends to maintain the following dynamic relationship expressed by

$$x_r - x = F\left(D\right) f_e \quad (23)$$

where x_r stands for the point-to-point reference trajectory; x denotes the current position of the end-effector in operational space; $F(D)$ is a stable, second-order linear filter described by

$$F\left(D\right) = K_d\left[D^2M_d + DB_d + K_d\right]^{-1}, \ D = \frac{d}{dt} \quad (24)$$

whose stiffness, damping, and inertia matrices, $K_d = diag\{k_d,\ldots,k_d\}$, $B_d = diag\{b_d,\ldots,b_d\}$, $M_d = diag\{m_d,\ldots,m_d\}$, are of m-order, diagonal and positive-definite, and with the same diagonal entries. Hence, with the dynamic relationship in (*), the impedance error is defined by

$$\tilde{\xi} = x_r - x - F(D) f_e \tag{25}$$

Theoretically, the goal of the proposed impedance control method is to make the impedance error convergent to zero after a long time, i.e., $\lim_{t\to\infty} \tilde{\xi} = 0$. It is worthwhile to note that the impedance control become position control if the constraint force f_e is zero. Therefore, the proposed impedance control scheme can be regarded as a position-control approach allowing a tolerant error in the reference trajectory tracking in the presence of innegligible interaction forces. To show the boundedness and elimination of the steady-state impedance error in Figure 4, one obtains

$$x_d = x_r - F(D) f_e + K_p \tilde{\xi} + K_i \int_0^t \tilde{\xi}(\tau) d\tau \tag{26}$$

Via the IK solver, the desired joint angles q_d is given by

$$\begin{aligned} q_d &= K^{-1}(x_d) \\ &= K^{-1}\left[x_r - F(D) f_e + K_p \tilde{\xi} + K_i \int_0^t \tilde{\xi}(\tau) d\tau \right] \end{aligned} \tag{27}$$

where K_p and K_i are diagonal and positive-definite. By defining the joint angle error as $\tilde{q} = q_d - q$, we have $q = q_d - \tilde{q}$, and then use the nonlinear forward kinematic mapping and Taylor series expansion to attain

$$\begin{aligned} x &= K(q) = K(q_d - \tilde{q}) \\ &= K\left(K^{-1}\left[x_r - F(D) f_e + K_p \tilde{\xi} + K_i \int_0^t \tilde{\xi}(\tau) d\tau \right]\right) - \tilde{x} \\ &= x_r - F(D) f_e + K_p \tilde{\xi} + K_i \int_0^t \tilde{\xi}(\tau) d\tau - \tilde{x} \end{aligned} \tag{28}$$

where $\tilde{x}$ denotes the position error in operational space. Thus, it is easy to show that the position error is bounded and even zero since from (28) we have

$$\tilde{x} = (I + K_p)\tilde{\xi} + K_i \int_0^t \tilde{\xi}(\tau) d\tau \tag{29}$$

which leads to $\tilde{\xi} = (I + K_p)^{-1} \tilde{x}$ and $\|\tilde{\xi}\| \le \left\| \left[I + K_p \right]^{-1} \right\| \|\tilde{x}\| < \|\tilde{x}\|$ if only the proportional gain is used. Obviously, the steady-state impedance error will be eliminated by the integrator if the closed-loop system is still maintained stable.

B. PID Torque Control

This subsection is devoted to developing six independent PID controllers for the inner position control loops, as shown in Figure 4. Since a motor driver with the torque control mode is used for each joint motor control, the mathematical model of each joint under torque-based position control is modeled by the following second-order system model

$$G_i(s) = \frac{1}{s(J_i s + B_i)} \tag{30}$$

where J_i and B_i are respectively the moment inertial and viscous parameter of the i^{th} joint. To achieve the position control for the model in (30), it is effective to propose a PI controller with a desired second-order characteristic equation, $s^2 + 2\zeta_i \omega_{ni} + \omega_{ni}^2 = 0$, where ζ_i and ω_{ni} are respectively the desired damping ratio and natural frequency for each joint control system. The PD parameters of each controller are found by $K_{Pi} = J_i \omega_{ni}^2$ and $K_{Di} = 2\zeta_i J_i \omega_{ni} - B_i$ for i=1,…,6. Since the only one undesired zero will cause the desired transient response for each joint angle control, a prefilter with the transfer function

$$G_{pi}(s) = \frac{K_{pi} / J_i}{(K_{Di} / J_i)s + K_{Pi} / J_i} \tag{31}$$

is designed to eliminate such an unwanted zero response. Worthy of mention is that the PD controller for each joint will also eliminate the effect of the constraint force from the environment interaction, thus providing consistent control performance for each joint angle control loop.

VI. Force-Based Teaching

This section is devoted to describe the force-based teaching method by finding the torque components respectively for forward dynamics due to motion and gravity, friction torque due to static friction and inward kinematics due to the input force, and. To this end, each link of a manipulator is regarded as a rigid body. Once the locations of the center of mass and the inertia tensor of the link have been known, its mass distribution will be completely characterized. In order to move these links, one has to accelerate and decelerate them. The forces required for such motion are functions of the desired accelerations as well as the mass distributions of the links. In doing so, Newton's equation, along with its rotational analogue, Euler's equation, is employed to describe how forces, inertias, and accelerations are related to joint torques in all the six joints. Notice that torque components calculated from the forward dynamics will become the gravity compensation quantity if all the speed and acceleration information is set by zero.

A. Forward Dynamics

In the subsection, the forward kinematics is computed by an iterative Newton-Euler algorithm. By assuming that there is a rigid body rotating with angular velocity ω and angular acceleration $\dot{\omega}$, this algorithm is interpreted in the following. Under this situation, the moment N, which must be acting on the body to cause this motion, is given by Euler's equation

$$\mathrm{N} = {}^{\mathrm{C}}\mathrm{I}\dot{\omega} + \omega \times {}^{\mathrm{C}}\mathrm{I}_{\omega} \tag{32}$$

where ${}^{\mathrm{C}}\mathrm{I}$ is the inertia tensor of the body with respect to a fixed frame, $\{\mathrm{C}\}$, where origin is located at the center of mass.

We now consider the problem of computing the torque vector that corresponds to a given trajectory of a manipulator. We assume that the position, velocity, and acceleration vectors of the joint angles, $(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}})$, are given, where $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$. With this knowledge, and with knowledge of the kinematics and the mass-distribution information of the manipulator, one calculates the joint torque values for desired motion.

In order to compute inertial forces exerting on the links, it is necessary to compute the rotational velocity and linear and rotational acceleration of the center of mass of each link of the manipulator at any given instant. These computations will be done in an iterative way, starting with link 1 and moving successively, link by link, outward to link n.

$$^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\,{}^{i}\omega_i + \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{33}$$

From (9), one obtains the equation for transforming angular acceleration from one link to another:

$$^{i+1}\dot{\omega}_{i+1} = {}^{i+1}_{i}R\,{}^{i}\dot{\omega}_i + {}^{i+1}_{i}R\,{}^{i}\omega_i \times \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} + \ddot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{34}$$

The linear acceleration of each link-frame origin is obtained by

$$^{i+1}\dot{v}_{i+1} = {}^{i+1}_{i}R\left[{}^{i}\omega_i \times {}^{i}P_{i+1} + {}^{i}\omega_i \times \left({}^{i}\omega_i \times {}^{i}P_{i+1}\right) + {}^{i}\dot{v}_i\right] \tag{35}$$

Here, we imagine the frame, {C$_1$}, attached to each link, having its origin located at the center of mass of the link and having the same orientation as the link frame, {i}. Since equation (7) doesn't involve the joint motion at all and so is valid for joint (*i*+1), regardless of whether it is revolute or prismatic.

Having computed the linear and angular accelerations of the mass center of each link, we apply the Newton-Euler equations to compute the inertial force and torque acting at the center of mass of each link. Thus we have

$$\begin{aligned} F_i &= m\dot{v}_{C_i} \\ N_i &= {}^{C_i}I\dot{\omega}_i + \omega_i \times {}^{C_i}I\omega_i \end{aligned} \tag{36}$$

Note that the torque is reduced to the gravity compensation one in the static case where all the joints are stopped at some fixed position and posture.

B. Friction Compensation

Static friction in each joint can be directly estimated by maintaining the manipulator at its vertical condition. Under such a condition, gravity effect on each joint can be neglected. Therefore, the static friction in each joint is straightforward measured by increasing the applied torque from zero with a small step size, for example 0.1% of full-scale torque input. Since the static friction is assumed to be independent of any position and posture, the estimated friction torque is effective in compensating for the actual frictions.

C. Inward Dynamics

The inward dynamics is to compute the resultant torque for each joint if the input force at the end effector is measured by the 6-DOF force/torque sensor. Once the forces and torques acting on each link have been computed, we now need to calculate the joint torques that result in these net forces and torques being applied to each link. This can be done by writing a force-balance and moment-balance equation based on a free-body diagram of a typical link. Each link has forces and torques exerted on it by its neighbors and in addition experiences an inertial force and torque. To attain the goal, we define special symbols for the force and torque exerted by a neighbor link, which we repeat here:

f_i = force exerted on link i by link $i-1$,

n_i = torque exerted on link i by link $i-1$.

By summing the forces acting on link i , we arrive at the force-balance relationship:

$$F_i = {}^{i}f_i - {}_{i+1}^{\;\;i}R^{i+1}f_{i+1} \tag{37}$$

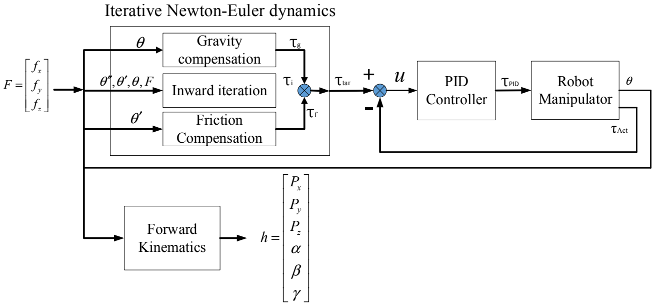


Figure 5. Block diagram of the force compliance teaching in the easy and versatile manipulation system.

By summing torques about the center of mass and setting them equal to zero, we obtain the torque-balance equation:

$$^{i}N_i = {}^{i}n_i - {}^{i}n_{i+1} + \left(-{}^{i}P_{C_i}\right) \times {}^{i}f_i - \left({}^{i}P_{i+1} - {}^{i}P_{C_i}\right) \times {}^{i}f_{i+1} \tag{38}$$

Finally, we rearrange the force and torque equations so that they appear as iterative relationships from higher numbered neighbor to lower numbered neighbor:

$$\begin{aligned} {}^{i}f_i &= {}_{i+1}^{\;\;i}R^{i+1}f_{i+1} + F_i \\ {}^{i}n_i &= {}^{i}N_i + {}_{i+1}^{\;\;i}R^{i+1}n_{i+1} + {}^{i}P_{C_i} \times {}^{i}F_i + {}^{i}P_{i+1} \times {}_{i+1}^{\;\;i}R^{i+1}\,{}^{i}f_{i+1} \end{aligned} \tag{39}$$

These equations are evaluated link by link, starting from link i and working inward toward the base of the robot. As in the static case, the required joint torque values are found by taking the Z component of the torque applied by one link on its neighbor:

$$\tau_i = {}^{i}n_i^T\,{}^{i}\hat{Z}_i \tag{40}$$

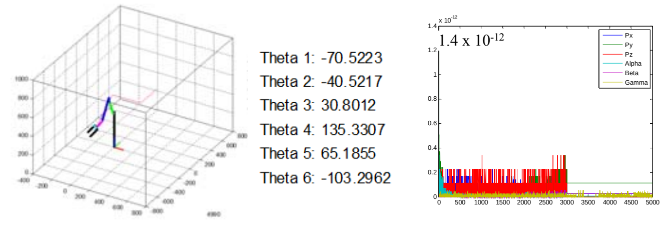
If the robot is in contact with the environment, the forces and torques due to this contact can be included in the force balance by having nonzero input force $^{N+1}f_{N+1}$ and the input torque $^{N+1}n_{N+1}$.

D. Torque-Based Motor Driving

This subsection is dedicated to develop six independent PID torque controllers for the force compliance teaching loops, as shown in Figure 5. Since a motor driver with the torque control mode is used for each joint motor control, the mathematical model of each joint under torque-based position control is modeled by the following two-order system model

$$G_i\left(s\right) = \frac{1}{s(J_i s + B_i)} \tag{41}$$

where J_i and B_i are respectively the moment inertial and viscous parameter of the i^{th} joint. To achieve the force compliance teaching for the model in (41), it is reasonable to assume that the torque control loop using the PID controller has a relative short time constant I comparison with the mechanical time constant. Hence, the dynamic behavior of the PID torque control loop can be ignored, and equation (41) is directly applied to attain the time response of each motor, thereby gives the time evolutions of all the six joint angles. As a result, the position and posture of the manipulator can be obtained due to its forward kinematics.



(a) (b)

Figure 6. (a) Selection of testing poses; (b) Errors between the true and resultant poses.

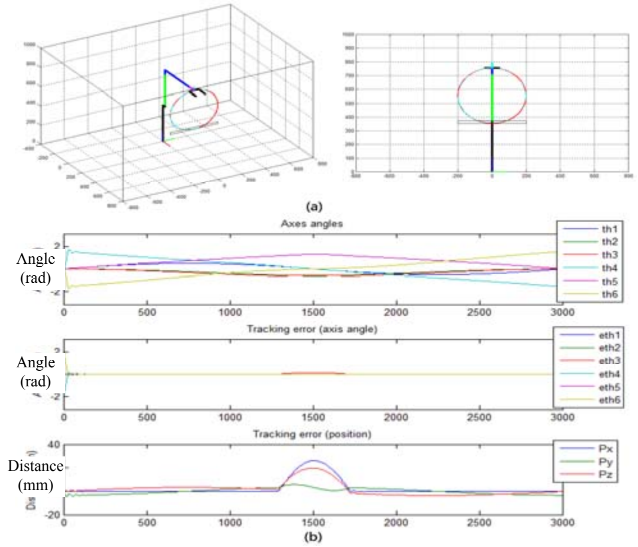


Figure 7. (a) Simulation of the circular trajectory tracking with the proposed P impedance controller; (b) Position tracking error while using the PI impedance controller.

VII. Simulation and Discussion

This section will conduct three simulations to examine the feasibility and effectiveness of the proposed inverse and forward kinematics, PI kinematic impedance methods and force-based teaching. Given the target positions and orientations, the first simulation is aimed to find all the six joint angles according to the algorithm in Section IV, and then apply the forward kinematics described in Section III to find all the positions of the six links and plot these positions for inspection purpose. The second simulation is devoted to verifying the performance and merit of the proposed PI kinematic impedance control method based on the aforementioned forward and inverse kinematics equations. The third simulation is aimed to find the torque outputs of all the six joints according to the methods in Section VI. Four simulations are done using Matlab/Simulink, and the DH parameters listed in Table 1 and the following mass parameters: m_1=20 kgw; m_2=6.485 kgw; m_3=15.4175 kgw; m_4=4.1522 kgw; m_5=3.9558 kgw; m_6=1.4285 kgw. Besides, all the friction torques are assumed constant and equal to 0.3 N-m.

A. Verification the proposed forward and inverse kinematics equation

The first simulation is conducted to validate both proposed analytical inverse and forward kinematics methods for the working space of the HIWIN robotic manipulator except singular points. Figure 6(a) depicts a set of testing poses of the manipulator. During the simulation, all the six joint angles are solved by the proposed inverse kinematics equation, the corresponding resultant poses are found by using the forward kinematics equations, and, finally, the average error between the desired and resultant poses is computed. Consequently, as shown in Figure 6(b), the average error is less than 1.4×10^{-12}mm and the average error is less than 0.2×10^{-12}mm, which is less than the accuracy or repeatability of the manipulator.

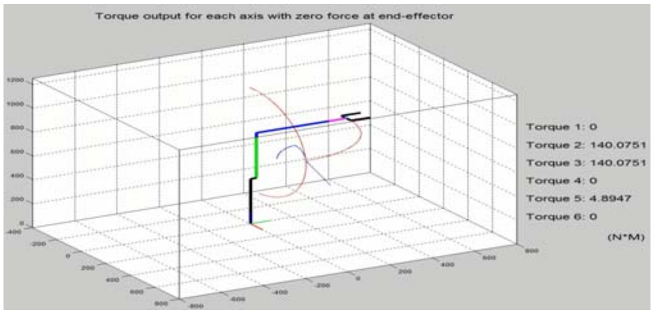


Figure 8. Testing continuous trajectories of the manipulator for force compliance teaching .and

B. Verification of the proposed PI kinematic impedance control method

The second simulation is carried out to examine the effectiveness of the proposed PI kinematic impedance control method. In doing this simulation, the environment stiffness is $K_e=diag\{0.5,0.5,0.5\}$ (N/m), the diagonal entries of the second-order linear filter are given by K_d=100, b_d=100 and m_d=100 for the resultant damping ratio 0.5, and the simulation time is 3 seconds. Figure 7(b) shows the position errors in the circular trajectory tracking simulation when the only PI impedance control gain is set by $K_P=diag\{0.75,0.75,0.75\}$ and $K_I=diag\{0.01,0.01,0.01\}$. The results in Figure6 reveal that the position errors were small and the manipulator moved along the environment while touching it, and the impedance errors were smaller than the position errors. This result confirms the effectiveness of the proposed control method.

C. Verification of the proposed the proposed force -based teaching.

The third simulation is conducted to validate the proposed torque generation method for force compliance teaching of the HIWIN robotic manipulator. Figure 8 depicts the testing continuous trajectories of the manipulator. During the simulation, the input force is assumed to be $[0,0,0]^T$ (unit: N-m), all the velocities and accelerations of all the six joints are set by zero. Thus, the torque outputs of the six joints are iteratively calculated via the Newton-Euler forward dynamic equations along with the inward equations. This result confirms the feasibility of the proposed force compliant teaching method.

VIII. Experimental Results and Discussion

This section aims to examine the effectiveness of the proposed PI kinematic impedance control method and force-based teaching which were implemented with the real industrial robotic manipulator.

A. Verification of the proposed kinematics-based impedance control method

Figure 9(a) shows the experimental equipment and environment, and Figure 9(b) shows the test results of the proposed impedance control where the target force was 19.61N (2kgw) and average force readings equaled 20.74N (2.1162kgw). This experiment was conducted by moving the end-effector (equipped with the 6-axis force/torque sensor) along the Z-axis to reach a fixed goal, force it to press the electronic load-cell scale at a desired force and then observing and taking the force measurements. The experimental data show that the average force readings are very close to the desired force imposed on the electronic load-cell scale, namely that the proposed method is capable of achieving impedance control so as to keep target force and target position (keep contact) as predicted simultaneously.

B. Experimental verification of the proposed force -based teaching.

The following two experiments are conducted to examine the effectiveness of the proposed method with the friction compensation, gravity compensation and inward dynamics using the easy and versatile manipulation system where the robotic manipulator is operated using the torque control mode. In the first experiment, the inward dynamics is not used. In order to achieve the friction compensation and gravity compensation, the actual position from EtherCAT driver is used to determine the size of the compensated friction in each joint. With friction and gravity compensations, the user pushed the manipulator by one finger easily as Figure 10 shows. The experimental results show the effectiveness of both friction and gravity compensations at two different postures. The second experiment was performed to show the merit of the proposed teaching method with both friction compensation and gravity compensations, and the inward dynamics in which the input force comes from the 6-DOF force/torque sensor. Figure 11 displays two still pictures which illustrate how the user easily moves the manipulator by hand with very light force.

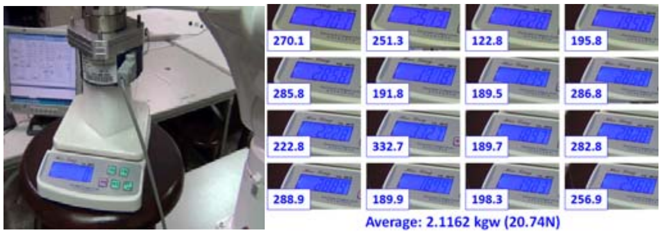


Figure 9. Experimental Set-up: (a) the experimental electronic load-cell scale for the proposed system; (b) experimental result of the proposed system.

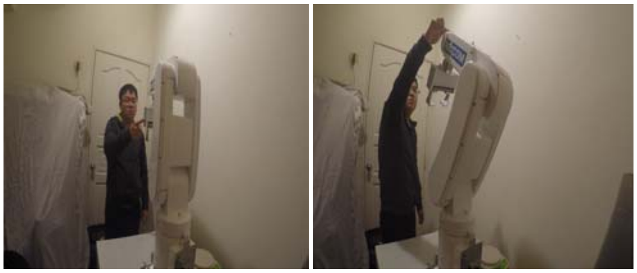


Figure 10. Two still pictures of the HIWIN manipulator moved by using only one finger when only both friction compensation and gravity compensation methods are used.

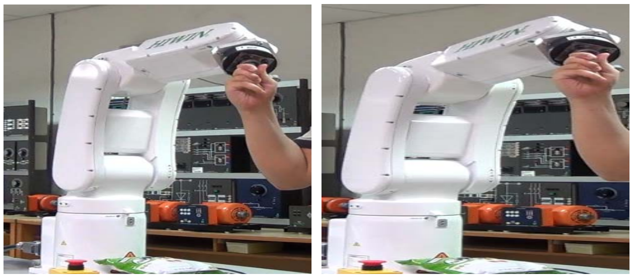


Figure 11. Two still pictures of the HIWIN manipulator moved by using the proposed torque generation method with the inward dynamics and the 6-DOF force/torque sensor.

IX. Conclusions and Future Work

This paper has presented an EtherCAT-based impedance control method and a force compliance teaching for a 6-DOF industrial robotic manipulator. In order to achieve the impedance control and force compliance teaching goals, this paper introduced the overall system structure of this impedance controller using EtherCAT protocol. The forward kinematics of the manipulator has been established using its DH parameters, and the analytical inverse kinematics method has also been proposed to solve the joint angles of the manipulator for any pose except singular points. The proportional-integral (PI) impedance controller along with a 6-DOF force/torque sensor and six independent proportional–integral–derivative (PID) torque controllers have been presented to achieve impedance control with satisfying performance. The torque generation method based on the Newton-Euler forward dynamics and inward dynamics has been proposed to obtain the required torque for each joint if the applied force in each joint is measured by the 6-DOF force/torque employing the experimental method, and the torque with the computed torque is shown effective in moving

the manipulator along the direction of the applied force. The overall force compliance teaching method with the PID torque-based motor driving schemes has also been shown useful in achieving force compliance teaching. Through the simulation results on a 6-DoF industrial manipulator HIWIN RA605, the effectiveness and merits of the proposed inverse kinematics method, impedance controller and force compliance teaching method have been confirmed in achieving position and impedance control. An interesting topic for future work would to seek for real applications by using the proposed PI-based impedance control for the manipulator.

References

[1] I. Jung and S. Lim, “An EtherCAT based control system for human-robot cooperation,” in Proc. of 2011 16th Intern. Conf.on Methods and Models in Automation and Robotics (MMAR), pp.341-344, 22-25 Aug. 2011.

[2] I. Jung and S. Lim, “An EtherCAT based real-time centralized soft robot motion controller,” in Proc. of 2012 Intern. Symp. on Instrumentation & Measurement, Sensor Network and Automation (IMSNA), vol.1, pp.117-120, .25-28 Aug. 2012.

[3] D. Orfanus, R. Indergaard, G. Prytz and T. Wien, “EtherCAT -based platform for distributed control in high-performance industrial applications,” in Proc. of 2013 IEEE 18th Conference on Emerging Technologies & Factory Automation, pp.1-8, 10-13 Sept. 2013.

[4] G. Zeng and A. Hemami, “An overview of robot force control,” Robotica, vol. 15,no. 5, pp 473-482,1997.

[5] B. Siciliano and L. Villani, Robot Force Control, Kluwer Academic Publishers, Boston, 2001.

[6] F. Caccavale , C.Natale, B. Siciliano and L. Villani,” Integration for the next generation: Embedding force control into industrial robots,” IEEE Robotics & Automation Magazine, vol.12, no.3, pp.53-64, Sept. 2005.

[7] R. V. Patel, J. Jayender and F.Shadpey, “A robust position and force control strategy for 7-DOF redundant manipulators,” IEEE/ASME Transactions on Mechatronics, vol. 14, no. 5, October 2009.

[8] O. M. Al-Jarrah and Y. F. Zheng,” Intelligent compliant motion control” IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol28., no.1, pp.116-122, Feb 1998.

[9] I. Bonilla, M. Mendoza, E. J. Gonz´alez-Galv´an, C. Ch´avez-Olivares, A. Loredo-Flores,and F. Reyes,” Path-tracking maneuvers with industrial robot manipulators using uncalibrated vision and impedance control,” IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol.42, no.6, pp.1716-1729, Nov. 2012.

[10] S. Jung, T. C. Hsia and R. G. Bonitz, “Force tracking impedance control of robot manipulators under unknown environment,” IEEE Transactions on Control Systems Technology, vol.12, no.3, pp.474-483, May 2004.

[11] C. C. Tsai, P. R. Deng, C. C. Chan, C. A. Lin, “Fast Position and Posture Control of an Anthropomorphous 7 DOF Dual-Arm Mobile Robot,” Proc. of 2013 CACS Intern. Automatic Control Conf., Sun-Moon Lake, Nantou, Taiwan, pp.227-232, Dec. 2-4, 2013.

[12] C. C. Cheah and D. Wang,” Learning impedance control for robotic manipulators,” in Proc. of 1995 IEEE International Conference on Robotics and Automation, 1995. Proceedings, vol. 2, pp2150-2155, 21-27 May 1995.

[13] S. Jung and T. C. Hsia,” Neural network impedance force control of robot manipulator,” IEEE Transactions on Industrial Electronics, vol. 45, no. 3, pp. 451-461, Jun 1998.

[14] T. Tsuji and K. Ito, “Neural network learning of robot arm impedance in operational space,” IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol.26, no.2, pp.290-298, Apr 1996.

[15] F. Caccavale, P. Chiacchio and A. Marino,” Six-DOF impedance control of dual-arm cooperative manipulators,” IEEE/ASME Transactions on Mechatronics, vol.13, no.5, pp.1083-4435, Oct. 2008.

[16] M. K. Wu, Y.S. Kung, Y.H. Huang, and T. H. Jung, “Fixed-point computation of robot kinematics in FPGA” in Proc. of 2014 Intern. Conf. on Advanced Robotics and Intelligent Systems (ARIS 2014), Taipei, Taiwan, 2014.

[17] S. Kucuk and Z. Bingul, “The inverse kinematics solutions of industrial robot manipulators,” Proceedings of the IEEE Intern. Conference on Mechatronics, ICM04, pp. 274-279, 2004.

[18] J.Zhao and N. I. Badler, “Inverse kinematics positioning using nonlinear programming for highly articulated figures,” ACM Transactions on Graphics, vol. 13, no.4, pp. 313-336, 1994.

[19] H. C. Huang and S. C. Lin “Hybrid GA-PSO Algorithm for Inverse Kinematics of 7-DOF Robot Manipulators,” Proceedings of 2011 Intern. Conf. on Service and Interactive Robots, National Chung Hsing University, Taichung, Taiwan, Nov.25-27, 2011.

[20] B. Fagin, “Ada/Mindstorms 3.0,” IEEE Robotics & Automation Magazine, vol. 10, no.2, pp. 19-24, 2003.

[21] T. Hasegawa, T.Suehiro, K.Takase, “A model-based manipulation system with skill-based execution,” IEEE Transactions on Robotics and Automation, vol. 8, no. 5, pp. 535-544, 1992.

[22] M. Skubic, R. A. Volz, “Acquiring robust, force-based assembly skills from human demonstration,” IEEE Transactions on Robotics and Automation, vol. 16, no.6, pp. 772-781, 2000.

[23] S. Ahmad,” A laboratory experiment to teach some concepts on sensor-based robot assembly systems,” IEEE Transactions on Education, vol. 31, no.2, pp. 74-84, 1988.

[24] J. L. Patton, F. A. Mussa-Ivaldi, “Robot-assisted adaptive training: custom force fields for teaching movement patterns,” IEEE Transactions on Biomedical Engineering, vol. 51, no. 4, pp. 626-646, 2004.

[25] G. Hirzinger, “Robot Systems Completely Based on Sensory Feedback,” IEEE Transactions on Industrial Electronics, vol. IE-33, no.2, pp. 105-109, 1986.

[26] R. Gassert, J. Metzger, K. Leuenberger, W. L. Popp, M. R. Tucker,B. Vigaru, R. Zimmermann, O. Lambercy, “Physical Student–Robot Interaction With the ETHZ Haptic Paddle, ” IEEE Transactions on Education, vol. 56, no.1, pp. 9 – 17, 2013.

[27] S.H.L. McAmis, K.B. Reed, “Simultaneous Perception of Forces and Motions Using Bimanual Interactions,” IEEE Transactions on Haptics,, vol. 5, no.3, pp. 220-230, 2012.

[28] K.Suwanratchatamanee, M. Matsumoto, S. Hashimoto, “Robotic Tactile Sensor System and Applications,” IEEE Transactions on Industrial Electronics, vol. 57, no.3, pp. 1074-1087, 2010.

[29] H. Asada, H. Izumi, “Automatic program generation from teaching data for the hybrid control of robots,” IEEE Transactions on Robotics and Automation, vol. 5, no.2, pp. 166-173, 1989.

[30] H.Koch, A. Konig, A. Weigl-Seitz, K. Kleinmann, J. Suchy, “Multisensor Contour Following with Vision, Force, and Acceleration Sensors for an Industrial Robot,” IEEE Transactions on Instrumentation and Measurement, Vol. 62, no.2, pp. 268-280, 2013.

[31] D. Powell, M. K. O'Malley, “The Task-Dependent Efficacy of Shared-Control Haptic Guidance Paradigms,” IEEE Transactions on Haptics, vol. 5, no.3, pp. 208-219, 2012.

[32] C. C. Tsai, F. C. Tai and C. A. Lin, “EtherCAT-based Impedance Control of a 6-DOF Industrial Robotic Manipulator,” in Proc. of 2015 National Symposium on System Science and Engineering, Tatung University, Taipei, Taiwan, 17~19 July, 2015.

Ching-Chih Tsai received the Diplomat in Electrical Engineering from National Taipei Institute of Technology, Taipei, Taiwan, ROC, the MS degree in Control Engineering from National Chiao Tung University, Hsinchu, Taiwan, ROC and the Ph.D degree in Electrical Engineering from Northwestern University, Evanston, IL, USA, in 1981, 1986 and 1991, respectively. Currently, he is currently a Distinguished Professor in the Department of Electrical Engineering, National Chung-Hsing University, Taichung, Taiwan, where he served the Chairman in the Department of Electrical Engineering from 2012 to 2014. He is a Fellow of IEEE, IET and CACS.

Dr. Tsai served as the Chair, Taipei Chapter, IEEE Control Systems Society, from 2000 to 2003, and the Chair, Taipei Chapter, IEEE Robotics and Automation Society from 2005 to 2006. In 2007, he was the program chair of 2007 CACS international automatic conference sponsored by Taipei chapter, IEEE control systems society. In 2010, he served as the program co-chair of SICE 2010 annual conference in Taiwan, which was technically sponsored by IEEE CSS; in 2011, he served as the General Chair, 2011 International conference on service and interactive robotics; in 2012, he has served as the General Chair, 2012 International conference on Fuzzy Theory and Its Applications, the General Chair, 2012-2015 CACS International Automatic Control Conferences, and the General Chair, 2016-2017 International Conference on Advanced Robotics and Intelligent Systems. Dr. Tsai served the two-term President, Chinese Institute of Engineers in Central Taiwan, Taiwan from 2007 to 2011, and two-term President of Chinese Automatic Control Society from 2012 to 2015. Since 2008, he has been the Executive Directors in Boards of Government of thee professional associations, including Robotic Society of Taiwan, Taiwan Fuzzy Systems Association, and Taiwan Systems Association. He has served as the Chair, Taichung Chapter, IEEE Systems, Man, and Cybernetics Society since 2009, the Chair of IEEE SMC Technical Committee on intelligent learning in control systems since 2009, the President of Robotics Society of Taiwan since 2016, the steering committee of Asian Control Association since 2014, a BOG member of IEEE Nanotechnology council since 2012, the Vice President of International Fuzzy Systems Association since 2015, and a BOG member of the IEEE SMCS since 2017.

Dr. Tsai has published more than 500 technical papers, and seven patents in the fields of control theory, systems technology and applications. Web of Science has indexed his paper entitled "Adaptive Neural Network Control of a Self-Balancing Two-Wheeled Scooter" in the category Automation Control Systems, where the paper was ranked 408 out of 37607 articles (published between 2010 to 2014). Dr. Tsai is respectively the recipients of the Third Society Prize Paper Award from IEEE Industry Application Society in 1998, the Outstanding Automatic Control Engineering Award in 2008 from Chinese Automatic Control Society (CACS), and the Outstanding Engineering Professor Award in 2009 from the Chinese Institute of Engineers in 2009, the IEEE Most Active SMC Technical Committee (TC) Award in 2012 from IEEE SMC Society, the Outstanding Electrical Engineering Professor Award from the Chinese Institute of Electrical Engineering in 2014, Outstanding Industry Contribution Award from Taiwan Systems Association in 2016, the best paper award in the International Journal of Fuzzy Systems in 2017, and many best paper awards from many international conferences technically supported by IEEE. He is the advisor, IEEE SMC student branch chapter at National Chung Hsing University; this chapter was the recipient of certificate of appreciation from IEEE SMCS in 2009. He has served as the associate editors of International Journal of Fuzzy Systems, and IEEE Transactions on Systems, Man and Cybernetics: Systems, IEEE Transactions on Industry Informatics, and International Journal of Electrical Engineering. Recently, he has served as the Editor-in-Chief of a new international robotics journal called "iRobotics". His current interests include advanced nonlinear control methods, deep model predictive control, fuzzy control, neural-network control, advanced mobile robotics, intelligent service robotics, intelligent mechatronics, intelligent learning control methods with their applications to industrial processes and intelligent machinery.

Chun-Chieh Chan received the M.S. degree in Department of Electrical Engineering from National Chung Cheng University, Chiayi, Taiwan, ROC. in 2007. His current research interests include computer vision, intelligent control, robot motion planning and their applications to human-robot collaboration and human-safe systems.

Chun-An Lin received the M.S. degree in Department of Electrical Engineering from National Chung Hsing University, Taichung, Taiwan, ROC. in 2015, respectively. His current research interests include intelligent control, robotic manipulator and their applications to industrial processes and machines.

Feng-Chun Tai received the B.S., M.S. and Ph.D. degrees in Department of Electrical Engineering from National Chung Hsing University, Taichung, Taiwan, ROC. in 2007, 2010 and 2018, respectively. His current research interests include mobile robots, intelligent control, navigation system and their applications to industrial processes and machines.

Mobile Robot Localization in Outdoor Environments by the Integration of GPS, IMU and Visual Odometry

Guo-Sheng Cai[1], Shih-Fen Kao[1], and Huei-Yung Lin[2]

Abstract—**In this work we present the localization and navigation for a mobile robot in the outdoor environment. It is based on fusing the data from IMU, differential GPS and visual odometry using the extended Kalman filter framework. First, the IMU provides the heading angle information from the magnetometer and angular velocity, and GPS provides the absolute position information of the mobile robot. The image-based visual odometry is adopted to derive the moving distance and additional localization information. Finally, the mobile robot position is further refined using the extended Kalman filter. The experiments are carried out in the outdoor environment. We compare the results with the original GPS raw data, and the performance of the presented method is evaluated.**

Index Terms—

I. Introduction

ONE of the primary technology for outdoor localization is using GPS. Although the location information can be obtained through GPS all year round, the signal transmission process is susceptible to interference [1]. This might be caused by several different reasons. Some common factors are the signals blocked by high-rise buildings, tunnels, elevated roads. Other obstacles in many areas can hinder the reception of GPS devices, and result in weak or no signals. In addition to the influence of the surrounding environment, the signals in the troposphere can also have some delays. Consequently, the signal generated error causes the time out of synchronization between the GPS receiver and the satellite, which leads to the decrease of the positioning accuracy [11].

To improve the accuracy of GPS for civilian users, the technology of Differential Global Positioning System (DGPS) has been developed [18]. With the positioning accuracy of about 3 meters, the reliability of GPS position is greatly enhanced. On the other hand, the development of autonomous driving has been a trend in recent years. Therefore, the research of robot navigation systems becomes an important topic, and the core technology is about the localization of mobile robots. Since the navigation system with multi-sensor integration has pros and cons, it is an interesting and challenging research direction. We can roughly know the location from the outdoor global positioning system (GPS), and the angular acceleration velocity and the direction information from the inertial navigation system (INS). The wheel encoder system and visual odometer, on the other hand, can provide the direction and distance of the movement to refine the position information. Thus, some common integration systems include VINS, INS/GPS [13], VO/GPS, odometry/GPS [10], odometry/INS [4], [21], which provide more reliable and accurate navigation information than a single sensor.

To improve the localization accuracy for mobile robot navigation, this paper presents a positioning method based on the Extended Kalman Filter (EKF) and implement on Robot Operating System (ROS). In the prediction step, the state of the robot is estimated by the optical sensor and the IMU. The predicted state is then corrected by a GPS measurement. To achieve a complementary relationship, on one hand we obtain more accurate position information, and on the other hand the cumulative error of visual odometry and IMU in a large-scale environment is improved.

II. Related Work

The visual odometry (VO) is to derive the 6-DOF camera pose using algorithms and the information collected by optical sensors [5]. It can be used for robot navigation, and the techniques are divided into monocular, stereo, and RGB-D approaches depending on what kind of sensor is adopted [8]. The main algorithm of VO framework usually includes feature point extraction, feature point matching, and feature point tracking. Two common types of motion models are used to estimate the camera pose. The first type is the sparse visual odometry, in which only a few feature points in the image are extracted, matched and tracked. The other technique is the dense visual odometry. Compared to the sparse approaches, it spends much more time due to the extraction of all feature points from the images. Nevertheless, it provides more reliable and accurate camera pose than the sparse visual odometry algorithms.

[1]Guo-Sheng Cai and Shih-Fen Kao are with Department of Electrical Engineering, National Chung Cheng University, 168 University Rd., Chiayi 621, Taiwan E-mail: 10101068a@gmail.com
[2]Huei-Yung Lin is with Department of Electrical Engineering, National Chung Cheng University, 168 University Rd., Chiayi 621, Taiwan and Advanced Institute of Manufacturing with High-Tech Innovation, National Chung Cheng University, 168 University Rd., Chiayi 621, Taiwan E-mail: lin@ee.ccu.edu.tw

The support of this work in part by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221- E-194-004 and the Advanced Institute of Manufacturing with High-tech Innovations (AIM-HI) from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan is gratefully acknowledged.

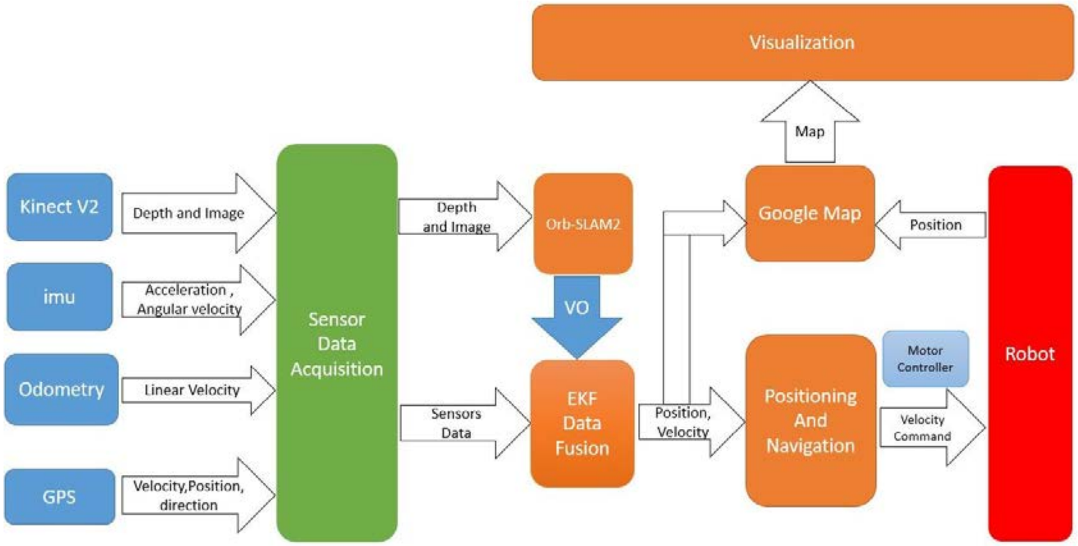


Figure 1. The system architecture for the proposed technique. The data are first collected from the sensors to derive the current approximate position and orientation. The robot pose is then estimated by the extended Kalman filter, and finally displayed on the map.

In addition to the classification by the amount of features for processing, VO techniques can also be divided into filtering based and graph optimization according to the multi-sensor fusion back-end approaches. The current pose of the filtering based methods is only related to the pose of the previous time frame, but it is related to all previous poses for the graph optimization methods. Based on the order of data integration among the sensors, the system flow can be either tightly coupled or loosely coupled. The advantage of loosely coupled is that the integration is simple, while the relatively complicated fusion method with tightly coupled can provide more accurate pose.

One of the sparse visual odometry is the feature-based method. It can be realized by low-cost hardware because of the low computation time requirement, and has been widely adopted in robots [9], self-driving vehicles [19] and UAVs [14]. The algorithms generally consist of the following steps: First, the effective feature points are extracted from the images. Second, the common feature points between the images are matched. Finally, the feature points are tracked by searching the image sequence to estimate the camera pose. The feature-based approaches only utilize part of the image, and the traditional methods for feature extraction including SIFT, SURF, ORB [3], [20]. Dryanovski *et al.* propose a real-time VO method which aligns 3D points against a consistent global model [6]. The camera pose is obtained by Kalman filter with dynamically updated observation information.

RGBD-SLAM is a feature-based (indirect) mapping system [7]. Qian and Jian use an environmental measurement model to verify the transformations estimated by feature correspondences and the ICP algorithm [17]. The pose map optimization and closed loop detection are also performed to improve the trajectory estimation. Recently, ORB-SLAM2 was released [15]. It is an open source SLAM system that uses ORB functionality for tracking, mapping and close-loop detection while running on a single CPU. Some outdoor navigation systems uses Kalman filter models to fusion GPS information and 6-axis inertial measurement data [16].

III. Exploration Strategy

When we refer to the outdoor positioning system, GPS is widely known to the public. The positioning accuracy of the civilian GPS is about 3 meters, which not is enough for the robot navigation purpose. For the current high-precision GPS, it is still too expensive to for most applications. In addition, GPS signals are susceptible in the outdoor environment, so there is a need to refer to other information sources. This paper presents a positioning system that fuses multiple sensor inputs. The system architecture is shown in Fig. 1. First, the data are collected from the sensors to derive the current approximate position and orientation. Next, the robot pose is estimated by the extended Kalman filter, and finally displayed on the map.

A. *Robot Localization System*

In recent years, SLAM research mostly adopts low-cost monocular cameras or stereo cameras with high precision measurement. A well-known example is the ZED stereo camera used for the outdoor environment. For the algorithms, ORB-SLAM2 is a feature-based SLAM system. It can be used with monocular, stereo and RGB-D cameras with high-precision positioning capability. For the development of RGB-D cameras, the depth measurement methods have been improved, from the early light coding to the recent time-of-flight (TOF) techniques. Thus, this work adopts the RGB-D camera as the front end of ORB-SLAM2, to provide the visual odometry information.

In ORB-SLAM2, the front end contains two threads, *tracking* and *local mapping*. The main tasks of the tracking thread include feature point extraction, camera pose estimation, tracking, and optimization, as well as the keyframe generation. On the other hand, the main job of the local

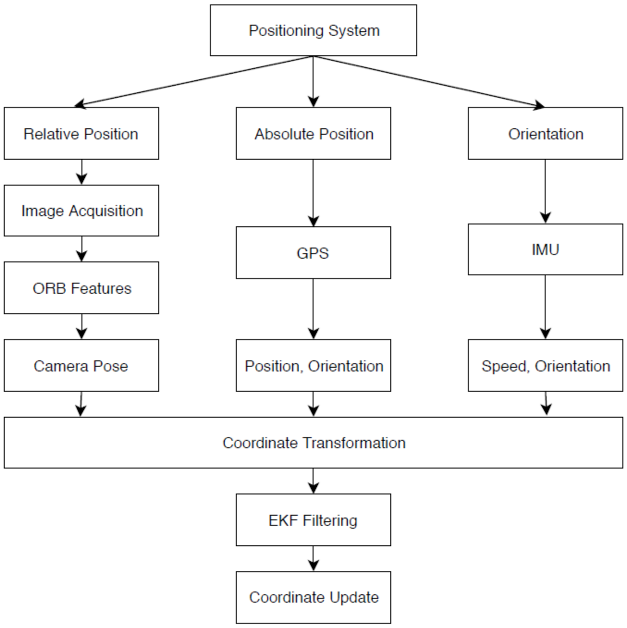


Figure 2. The flowchart with three sensor inputs (GPS, IMU, and camera) of the positioning system.

mapping thread is to calculate the descriptor of the new keyframe, insert the new keyframe to the map, delete the redundant points and key frames in the map, and finally obtain an optimized pose by bundle adjustment.

The positioning system flowchart is shown in Fig. 2. The GPS mainly provides the coordinates of absolute positions and the heading angles of the robot in the real world. The visual odometer extracts the feature points from the images, and then matches and tracks among frames to obtain the relative distance of the movement. The IMU sensor gives the acceleration and orientation of the robot. By processing the sensor inputs, the resulting information is converted to the same coordinate system. Finally, all of the data from different sensors are fused through the EKF to calculate a new position for the robot.

B. *Extended Kalman Filter*

The EKF algorithm adopted in this work is referenced from [12]. According to the statistical data processing and principles of modern control theory, estimating the system state vector from the measurement signal contains noise by data iteration [2]. Thus, the main algorithm is to estimate the optimal value of the current system state by the one estimated at the previous time instant and the current measurement. The state equation of a system can be expressed by

$$x_t = g(u_t, x_{t-1}) + \omega_{t-1} \tag{1}$$

where x_t is the robot's system state at time *t*, *g* is a nonlinear state transition function, u_t is a speed control function, and ω_{t-1} is the process noise. Our objective is to estimate the robot location in a 2-D environment, and move the robot to a goal position. Thus, the system is assumed to be normally distributed, and the 6-D state vector, *x*, consisting of the robot's position, 2-D orientation, and the velocities, or

TABLE I

THE EKF ALGORITHM FLOW.

Extended Kalman filter $(x_t, \Sigma_t, u_t, z_t)$
Prediction step:
1. $\bar{x}_t = g(u_t, x_{t-1})$
2. $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$
3. $R_t = \begin{bmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.3 \end{bmatrix}$
Update step:
4. $K_t = \bar{\Sigma}_t H_t^T (A_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$
5. $x_t = \bar{x}_t + K_t(z_t - h(\bar{x}_t))$
6. $\Sigma_t = (I - K_t H_t)\bar{\Sigma}_t$
7. $\text{return } x_t, \Sigma_t$

$$x_t = x_{t-1} - \frac{v}{\omega} sin\theta + \frac{v}{\omega} sin(\theta + w\Delta t) + w_{1,t} \tag{2}$$

$$y_t = y_{t-1} - \frac{v}{\omega} cos\theta + \frac{v}{\omega} cos(\theta + w\Delta t) + w_{1,t} \tag{3}$$

$$\theta_t = (\theta_{t-1} + \omega_{t-1}) + \omega_{3,t} \tag{4}$$

$$vx_t = v_t \cdot cos\,\theta + \omega_{4,t} \tag{5}$$

$$vy_t = v_t \cdot cos\,\theta + \omega_{5,t} \tag{6}$$

$$\omega_t = \omega_{t-1} + \omega_{6,t} \tag{7}$$

where x_t, y_t indicate the position at time *t*, θ_t is the orientation angle, ω_t is the angular velocity, and $w_{1,t}$ to $w_{6,t}$ denote the state noise. Additionally, we define the measurement model with

$$z_t = h(x_t) + v_t \tag{8}$$

where z_t is the measurement at time *t*, v_t is the measurement noise with a normal distribution, and *h* is a nonlinear function transforming the state to the measurement space.

In the algorithm, the current state vector and the prediction error covariance matrix are calculated by

$$\bar{x}_t = g(u_t, x_{t-1}) \tag{9}$$

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t \tag{10}$$

where g is a basic kinematic model derived from Newtonian mechanics. The estimate error covariances $\bar{\Sigma}_t$ projected by the function G, the Jacobian of G, and influenced by *R*, the covariance with additive white Gaussian noise to process noise.

Next, a correction step is given by

$$K_t = \bar{\Sigma}_t H_t^T (A_t \bar{\Sigma}_t H_t^T + Q_t)^{-1} \tag{11}$$

$$x_t = \bar{x}_t + K_t(z_t - h(\bar{x}_t)) \tag{12}$$

$$\Sigma_t = (I - K_t H_t)\bar{\Sigma}_t \tag{13}$$

where *K* is a matrix called the Kalman gain. It is calculated by

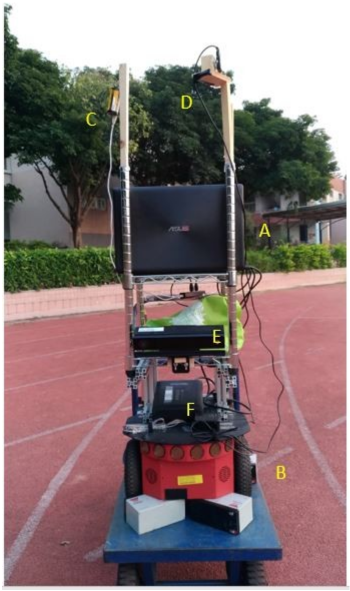


Figure 3. The equipment used in the experiments. It includes a laptop computer, a mobile platform, a GPS receiver, an IMU, an RGB-D camera, and an external power source.

TABLE II
LATITUDE AND LONGITUDE OF LANDMARK POINTS A TO H

landmark	Ground Truth	X-Y
A	120.481276, 23.55854	197045.24, 2606229.92
B	120.481362, 23.558432	197053.97, 2606217.92
C	120.481258, 23.558173	197043.25, 2606189.28
D	120.481144, 23.55791	197031.51, 2606160.2
E	120.480932, 23.557887	197009.86, 2606157.73
F	120.480851, 23.558014	197001.64, 2606171.83
G	120.480938,23.558243	197010.61,2606197.15
H	120.481073,23.558502	197024.5,2606225.79

the observation matrix, H_t, the measurement covariance, Q, and $\bar{\Sigma}_t$ in Eq. (10). The matrix Q is also additive white Gaussian noise and given by

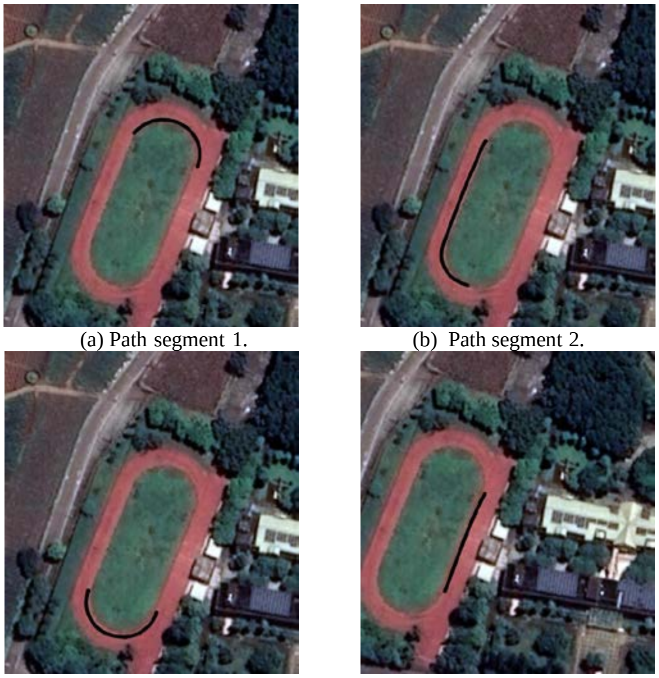
$$Q_t = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} \tag{14}$$

The gain is then used to update the state vector and the covariance matrix. Finally, x_t is the new best estimate, and this process can go on and feed x_t and Σ_t back to another round of prediction or update. However, when the number of iterations increases, it needs to spend more time to calculate.

The entire EKF algorithm flow is shown in Table I.

IV. EXPERIMENTAL RESULTS

Two experiments are carried out to evaluate the performance of the localization system. The software development environment is Ubuntu 16.04 and ROS. Pioneer-3DX is used as the mobile robot platform. The sensors adopted in this work including a 3DM-GX5-25 IMU, an RGB-D camera Microsoft Kinect V2, and a GPS receiver Holux M-241. As shown in Fig. 3, the sensors are mounted on a custom shelf to avoid magnetic interference for the IMU, reduce shaking for the camera, and increase signal quality for the GPS receiver. We use a laptop computer with an Intel Core i5-3230M CPU with 4GB memory to collect and process the sensor data.



(a) Path segment 1. (b) Path segment 2.

(c) Path segment 3. (d) Path segment 4.

Figure 4. Ground truth data collected for the experiments. The path consists of four segments for data collection.

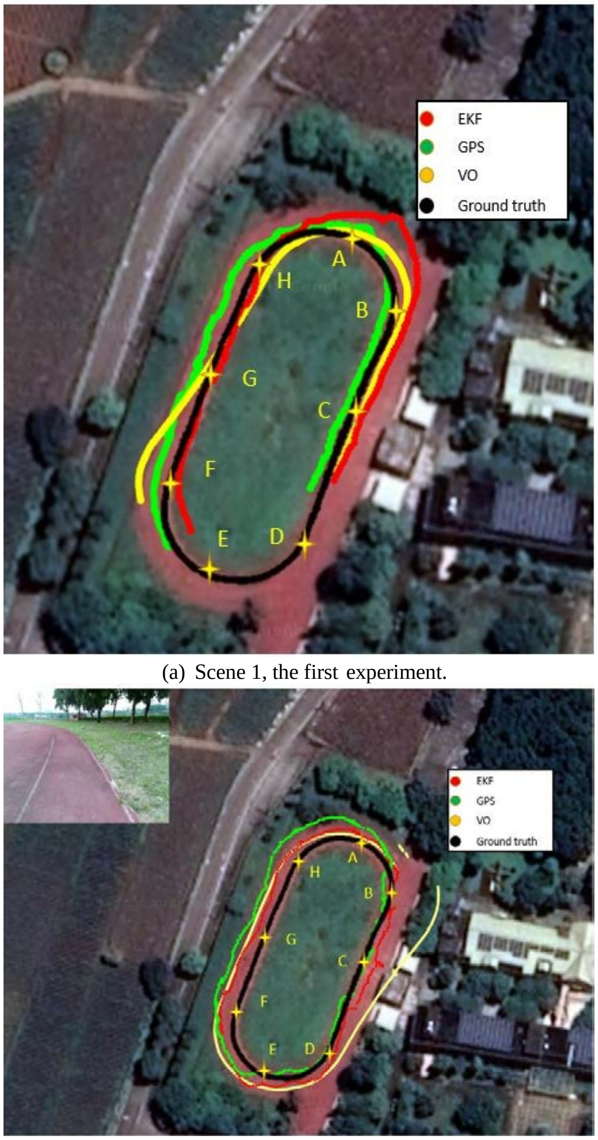
A. Ground Truth

Because the visual odometry has higher precision in short moving distances, in the experiments we divide the path into four parts, and collect the information for this four path segments with ORB-SLAM2. The results are shown in Fig. 4 using mapviz, a 2D visualization tool. Next, the complete route for ground truth consists of the four segments as the actual trajectory in the experiment. It is used to compare with the EKF positioning results. For comparison, we mark eight landmark points from A to H on the real track, and the latitude and longitude are shown in Table II.

B. Fusion GPS Data

In the experimental results shown in Figs. 5 and 6, the green curves represent the raw GPS data, the yellow curves represent the visual odometry measurements, the red curves represent the EKF results, and the real track is shown in the black curve. In the figures, we can easily observe the differences of the GPS measurements. The GPS signals in Scene 1 (see Fig. 5) has a higher accuracy. On the other hand, the GPS signals in Scene 2 (see Fig. 6) are occluded by buildings. The localization is not only caused the inaccurate GPS signal positioning, but also affected the visual odometry. The visual odometry is prone to provide the error information and loose track when the orientation is changed. However, it can still provide a reliable relative distance.

The reliable visual odometry in short distance not only helps to remove the GPS drift values for the EKF positioning algorithm, but also improves the estimation of the starting



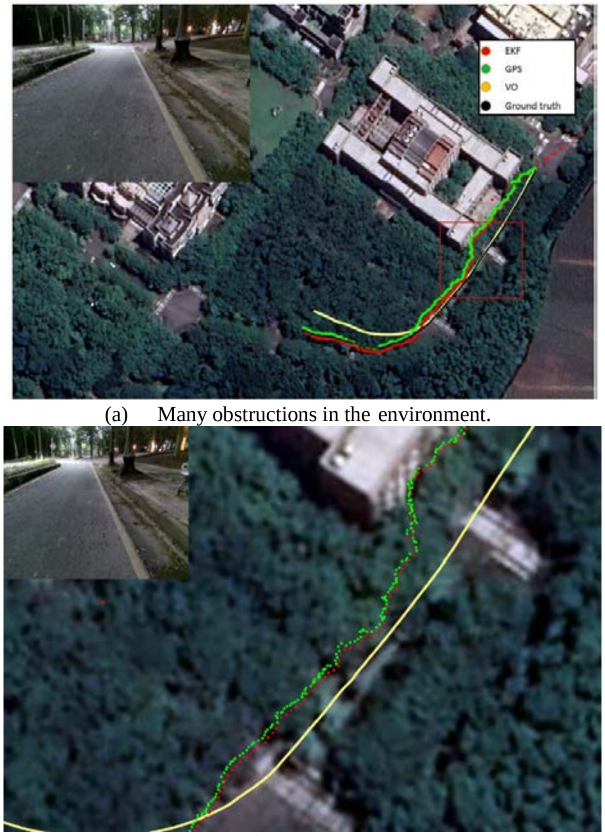
(a) Scene 1, the first experiment.

(b) Scene 1, the second experiment.

Figure 5. The experiments on Scene 1 with EKF: VO+IMU+GPS. Two experiments are carried out with the ground truth for comparison.

point. While the IMU can provides more accurate heading information for the robot. As for the GPS, it can maintain the consistency of the global map when the robot is moving in a large scale map.

The localization results are in general not good enough as illustrated. First, we collect all data and transform the GPS data to the Cartesian coordinate system. Next, we take the EKF results and raw GPS data to compare with the real trajectories, respectively. It is seen that the error rate is reduced from 79 to 6.6 meters (Table III) and from 4.48 to 3.7 meters (Table IV). In addition, we also use other methods for integration, such as IMU (see Fig. 7), VO and IMU (see Fig. 8). Due to the sensitivity and cumulative errors of the IMU, the results are not correct. Although the visual odometry can provide non-segment information, it is easily affected by the complex environment. Thus, it is difficult to be used for a long-term movement.



(a) Many obstructions in the environment.

(b) Partial zoom-in of the red rectangle in Fig. 6(a).

Figure 6. The experiments on Scene 2 with EKF: VO+IMU+GPS. Two experiments are carried out with the ground truth for comparison.

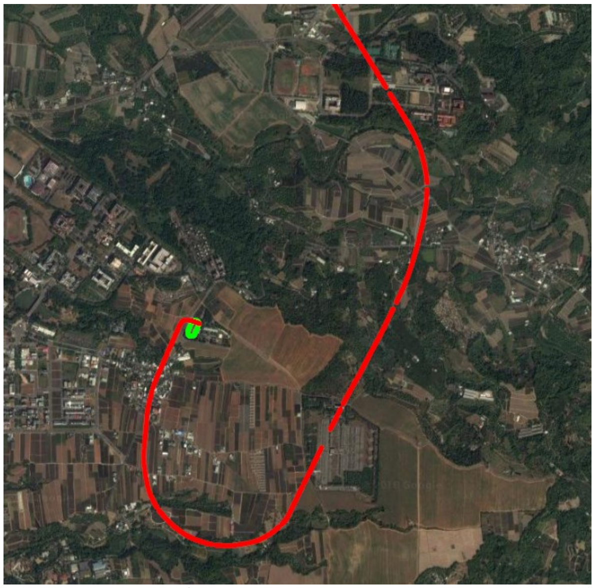


Figure 7. The trajectory from EKF with IMU. The result shows that IMU is not able to provide a reasonable trajectory.

V. CONCLUSION AND FUTURE WORK

In this paper, we present a real-time positioning system for an outdoor mobile robot. The system is tested in two outdoor scenarios. We use the absolute positions by GPS, the visual odometry information from the front-end of the ORB-SLAM2 algorithm, and the IMU to provide reliable direction

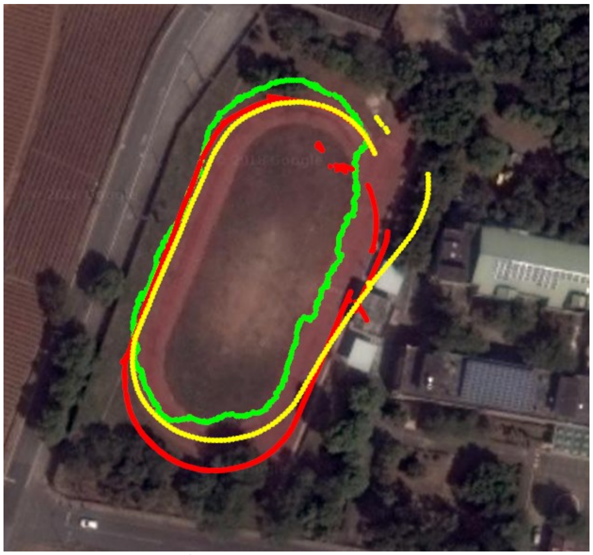


Figure 8. The trajectory from EKF with VO and IMU. It shows the results can be restricted locally.

TABLE III
THE RMS ERROR IN FIG. 5(A).

Error(m)	GPS	EKF
X direction	4.833333333	2.558333333
Y direction	77.27333333	1.973333333
ALL	79.68780111	3.745540089

TABLE IV
THE RMS ERROR IN FIG. 5(B).

Error(m)	GPS	EKF
start and end	4.223079445	1.893911296
X direction	4.48	3.805
Y direction	4.26875	1.88625

estimations to known the orientation of the robot movement. The sensor data are then fused by the EKF algorithm to obtain more accurate robot positions. The visual odometer gives a reliable message at the beginning, but as the moving distance increases, the error accumulation is caused by environment factors. AS a result, the distance and direction information becomes unreliable. Since the use of GPS information can limit the distance error to the range for keeping the trajectory consistent on the map, and the IMU can give the reliable direction information, we have fused the complementary information which improve the positioning accuracy accordingly.

REFERENCES

[1] M. Agrawal and K. Konolige, "Real-time localization in outdoor environments using stereo vision and inexpensive gps." In *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 3, pp. 1063-1068, Aug 2006.

[2] R. Brown and P. Hwang. *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises*. CourseSmart Series. Wiley, 2012.

[3] H. Chien, C. Chuang, C. Chen, and R. Klette. "When to use what feature? sift, surf, orb, or a-kaze features for monocular visual odometry." In *2016 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pp. 1-6, Nov 2016.

[4] B.-S. Cho, W.-s. Moon, W.-J. Seo, and K.-R. Baek, "A dead reckoning localization system for mobile robots using inertial sensors and wheel revolution encoding." *Journal of Mechanical Science and Technology*, vol. 25, no.11, pp. 2907-2917, Nov 2011.

[5] V. J. Delgado, P. R. Kurka, and E. Cardozo, "Visual odometry in mobile robots. In *IX Latin American Robotics Symposium and IEEE Colombian Conference on Automatic Control, 2011 IEEE*, pp. 1-4, Oct 2011.

[6] I. Dryanovski, R. G. Valenti, and Jizhong Xiao, "Fast visual odometry and mapping from rgb-d data." In *2013 IEEE International Conference on Robotics and Automation*, pp. 2305–2310, May 2013.

[7] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-d mapping with an rgb-d camera." *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 177-187, Feb 2014.

[8] J. Hsu and H. Lin, "Sparse edge visual odometry using an rgb-d camera. In *2017 11th Asian Control Conference (ASCC)*, pp. 964-969, Dec 2017.

[9] Jin-Hwan Joo, Dae-Han Hong, Yoon-Gu Kim, Ho-Geun Lee, Ki-Dong Lee, and Suk-Gyu Lee, "An enhanced path planning of fast mobile robot based on data fusion of image sensor and gps." In *2009 ICCAS- SICE*, pp. 5679–5684, Aug 2009.

[10] L. Li, Y. Liu, K. Wang, and M. Fang, "Estimating position of mobile robots from omnidirectional vision using an adaptive algorithm." *IEEE Transactions on Cybernetics*, vol. 45, no. 8, pp. 1633-1646, Aug 2015.

[11] H.-Y. Lin, C.-W. Yao, K.-S. Cheng, and V. L. Tran, "Topological map construction and scene recognition for vehicle localization." *Autonomous Robots*, vol. 42, no. 1, pp. 65-81, Jan 2018.

[12] T. Moore and D. Stouch, "A generalized extended kalman filter implementation for the robot operating system." In E. Menegatti, N. Michael, K. Berns, and H. Yamaguchi, editors, *Intelligent Autonomous Systems 13*, pp. 335-348, Cham, 2016. Springer International Publishing.

[13] Y. Morales, Eijiro Takeuchi, and Takashi Tsubouchi, "Vehicle localization in outdoor woodland environments with sensor fault detection." In *2008 IEEE International Conference on Robotics and Automation*, pp. 449-454, May 2008.

[14] T. Mouats, N. Aouf, L. Chermak, and M. A. Richardson, "Thermal stereo odometry for uavs." *IEEE Sensors Journal*, vol. 15, no. 11, pp. 6335-6347, Nov 2015.

[15] R. Mur-Artal and J. D. Tards, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras." *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255-1262, Oct 2017.

[16] E. North, J. Georgy, M. Tarbouchi, U. Iqbal, and A. Noureldin, "Enhanced mobile robot outdoor localization using ins/gps integration." In *2009 International Conference on Computer Engineering Systems*, pp. 127-132, Dec 2009.

[17] J. Qian and Y. Jiang, "Multi-type distributed generator planning considering locational marginal capacity cost of distribution network [j]." *Electric Power Automation Equipment*, vol. 36, no. 3, pp. 94-99, 2016.

[18] S. Rezaei and R. Sengupta, "Kalman filter-based integration of dgps and vehicle sensors for localization." *IEEE Transactions on Control Systems Technology*, vol. 15, no. 6, pp. 1080-1088, Nov 2007.

[19] D. Scaramuzza and R. Siegwart, "Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles." *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1015-1026, Oct 2008.

[20] C. Shan, H. Zhang, and Q. Xia, "Dealing with the structured scene in visual odometry(vo): Incomplete surf." In *2017 2nd International Conference on Robotics and Automation Engineering (ICRAE)*, pp. 436-440, Dec 2017.

[21] D. Wei, X. Ji, W. Li, H. Yuan, and Y. Xu, "Vehicle localization based on odometry assisted magnetic matching." In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1-6, Sep. 2017.

Guo-Sheng Cai receive his master degree in Electrical Engineering from National Chung Cheng University, Taiwan, in 2019. His research interests include image processing, computer vision, and robotics.

Shih-Fen Kao is currently a master student in Department of Electrical Engineering, National Chung Cheng University, Taiwan. His research interests include machine learning, computer vision, and advanced driving assistance systems.

Huei-Yung Lin received his BS degree in Applied Mathematics from National Chiao Tung University, Taiwan, and his MS and PhD degrees in electrical and computer engineering from the State University of New York at Stony Brook. In 2002 he joined the Department of Electrical Engineering, National Chung Cheng University, Taiwan, and currently is a full professor. His current research interests include computer vision, robotics, pattern recognition, and image processing. He serves as an organizing committee member of several robotics related IEEE international conference. He is also a founding member and deputy secretary general of Taiwan Society of Robotics.

Autonomous Social Robot with Moods

Zong-Ze Wu, Yu-Ting Hsiao, Xin-Yi Jian and Li-Chen Fu, Fellow, IEEE

***Abstract*—For a social robot to reach high autonomy, it should make its own decisions without users' command. Therefore, homeostatic drive theory and Maslow's hierarchy of needs have been adopted to model the robot, and we build the autonomous decision making system for robot to know what to do at every time moment. In order to pursue a better interaction between robot and human, it might be better for robot to possess personality and moods to some extent while interacting with human. Meanwhile, not only basic Question-Answering abilities but also abilities for Chit-Chatting should be considered crucial for the next level interactions. Therefore, in this research work, we build up a dialogue system for improving the relationship between robot and human. What's more, we develop a text style translation model to translate the style of the output sentence from chit-chat bot based on robot's moods. It is quite worthy to mention curiosity, which is important characteristic of human, and thus we develop visual question generation capability such that robot knows how to propose diverse questions to human for what it has observed. Finally, we evaluate our system separately for the three proposed modules, such as the autonomous system, text style translation module, and the visual question generation module.**

Results show that the autonomous system can make the robot satisfy the internal needs well and reach 99.98% in RSR index. In text style translation, the transferred sentiment accuracy in the Yelp dataset can reach 84.65%, which is an improvement. With the proposed evaluation metrics, which is designed for the visual question generation of our robot, we can say that robot will have a certain quality for this interaction.

***Index Terms*—Social Robot, Homeostasis Theory, Maslow's Hierarchy of Needs, Autonomous System for robot, Text Style Translation, Visual Question Generation**

I. Introduction

WITH the gradually aging society and lack of labor, the awareness of social robots accompanying elders and children rises. It is a very challenging problem, since the elders and children might prefer having a more humanlike robot rather than having a cold bloodless robot as friends. There are more and more social robots have been released, such as Jibo [1], Zenbo [2], Pepper [3], home service robots [4], etc. However, those robots are unlike task-specific robot, Roomba [5], which has specific goal for it to decide what to do at each moment. Therefore, in order to make the social robot not always wait for commands from human, we investigate homeostatic drive theory [6], which is the dominant method for social robots maintaining in a steady state, to improve the autonomy of it. Homeostatic drive theory produces drive to compensate the imbalanced internal needs of the agent. Maslow [7] extended the internal need of physiology to safety, love and belonging, esteem, and self-actualization for human. Therefore, inspired by Maslow, we designed our homeostatic model for social robots to make it more humanlike. After that, we adopt deep reinforcement algorithm for the autonomous decision making system to decide the action to balance the internal needs. The deep reinforcement learning takes advantage of making social robots possible to consider more complex situations and choose the actions more comprehensively.

In addition, the mechanism of moods of social robots is designed not only for more interesting and long-term interactions with humans [8], but also for showing its empathy while humans are in negative mood [9]. To express moods of robot, a text style translation model is deployed after getting the answering sentence from sequence to sequence chit-chat bot model, and it will translate the mood of the answering sentence. The question-answering abilities is also built for social robots being a good assistance to provide helps.

Moreover, if robot has its own curiosity after observing the environment and initiate the interaction by asking some questions related to what it observed, it can increase the connections and social interactions between the robot and human. In the field of human-robot interaction, researchers believe that taking initiative in interaction for robot can perform more fluent interactions [10][11]. Hence, a deep learning based visual question generation model is developed for proposing diverse questions with what the visual perception system of social robot has observed.

II. Related Works

There are three main parts that are presented in this paper, and therefore we will conduct the literature review separately for homeostatic model and decision making system, text style translation, and visual question generation.

A. Homeostatic Model and Decision Making System

There are many robots utilizing the concept of homeostasis to decide the action of robot, such as Sony's Aibo [12], MIT Kismet[13], NAO [14], etc. Cao et al. [15] proposed ROBEE architecture to generate drive from internal needs, and decide the action by preconditions-action pair for satisfying the highest drive. [16][17][18] follow similar formulation which modeled internal needs by drives and determine the intensity of motivation by drives and external stimuli, and the concept of motivation is introduced by Cañamero et al. [19]. Also, they utilized reinforcement learning for the decision making to learn

The authors are with the Advanced Control Laboratory, Graduate Institute of Electrical Engineering, National Taiwan University, Room 205, EE Building 2, No.1, Sec. 4, Roosevelt Road, Taipei, Taiwan 106 (e-mail: {r05921012, r06921010, d07921023, lichen}@ntu.edu.tw).

This research was supported by the Joint Research Center for AI Technology and All Vista Healthcare under Ministry of Science and Technology of Taiwan, and Center for Artificial Intelligence & Advanced Robotics, National Taiwan University, under the grant numbers of 108-2634-F-002-016 and 108-2634-F-002-017.

the relation between the dominant motivation and actions, and it is a similar way as in Gadanho et al. [20][21]. Lin et al. [16] and Yang et al. [17] used different ways to take human's intentions into consideration in the decision making system. Lin et al. [16] used the intention recognition for certain kinds of intention, and Yang et al. [17] designed some commands for users as the intention of users. However, because the dominant motivation is decided by whether the intensity of motivation is larger than a pre-defined threshold, which is a greedy way for determining the dominant one and it will directly affect the decision of action. In this work, deep reinforcement learning is utilized to consider all kinds of intensity of motivation, and it will help to target a more optimal long-term goal and consider more comprehensively. Different from the way of sensing the intention of human, we substitute it with an action of the robot, which is "standing by" and then the robot will wait for the user to interact while the action has been executed.

B. Text Style Translation

For the machine translation and summarization in text generation task, using large amount of parallel data has been important for deep learning method. However, there are not many parallel datasets for this task, and therefore, researchers has started focusing on learning the text generation from non-parallel data. The goal is to generate the sentence which preserves the content of the source sentence with the desired constraints.

Ficler et al. [22] proposed to use conditioned RNN language model, while desired semantic content and style will serve as conditional contexts. Han et al. [23] introduced two switches with seq2seq model to control the style of the sentence generation, helping the model to capture the semantic content in a style and to decode in a specific style. Mueller et al. [24] utilized a classifier to guide the modification of the latent representation in variational auto-encoder (VAE) and attained the sentence with desired attribute. Hu et al. [25] presented a new neural generative model combing VAE with adversarial training. VAE encodes the sentence into a latent representation and ties it with the style to generate specific style sentence. Shen et al. [26] introduced cross-alignment training to align both latent representation and sentence population with Professor-Forcing algorithm [27] and continuous relaxation for discrete sampling process [28], and they assumed the source and target domain sentence share the same latent space. Very similar to Shen et al. [26], Lample et al. [29] presented the method with alignment to the latent representation and two decoders are used to generate the sentence with specific styles. Zhao et al. [30] proposed style discrepancy loss and cycle consistency loss to handle the arbitrary style in the source domain, but a disadvantage is that it can only have one target style in target domain. Melnyk et al. [31] introduced the attention mechanism to the auto-encoder part in Shen et al. [26], and reduced the parameters for multiple styles by a collaborative classifier.

Our Text Style Translation module is largely influenced by Shen et al. [26], Zhao et al. [30], and Melnyk et al. [31]. We utilize a pre-trained language model for speeding up the convergence, and preserve better style information with an auto-encoder for style encoding. In addition, cycle consistency loss is also used to preserve a better semantic content, and N+1 discriminator [32] similar to collaborative classifier in Melnyk et al. [31], which is a good way to handle possible multiple moods for the robot in the future.

C. Visual Question Generation

There are several approaches for visual understanding and question generation, such as knowledge-based approach, data driven approach, etc. In knowledge-based approach, researchers utilize the information with high-level description in structured representation. Aditya et al. [33] integrated deep learning based perception module (object, scene, and scene constituent recognition) and the concept modeling from commonsense knowledge obtain from the text, and proposed scene description graph (SDG) to encode the relations among the entities from perception module. Then, they utilized the template based sentence generation (SimpleNLG [34]) to generate the sentence. Wu [35] also proposed to use deep learning perception module for high level structured information, and integrated the existing commonsense knowledge base (concept net) and the word embedding method to associate the structured information and topic database with different concept nodes for the following conversation. However, the limited database for the sentence generation is a big disadvantage in knowledge-based approach for the social robot, because human will quite easily to get tired of similar sentences that keep popping out. Data driven approach with deep learning does not suffer the same problem. Wu et al. [36] used external knowledge and encode it into vector for input in deep learning based sentence generation. For improving the various caption generation, Johnson et al. [37] presented Dense Captioning (DenseCap) which is similar to Faster R-CNN [38] and combines recognition network, localization network, and sentence generation, and it localizes and describes different regions of an image. Dai et al. [39] introduced conditional generative adversarial network with policy gradient for solving the image captioning. The generated sentences are more natural and diverse as compared to those from MLE-based model. Jain et al. [40] improved the diversity and creativity of the sentence generation by using variational auto-encoder approach. Based on Jain et al. [40], Wang et al. [41] proposed an additive Gaussian encoding space to the latent representation in variational auto-encoder.

Our Visual Question Generation module is highly inspired by Dai et al. [39] and Jain et al. [40], and we utilize variational auto-encoder with Gumbel-Softmax to for encoding phase of the latent representation for a more informative latent representation, compared with using conditional generative adversarial network, and introduce the discriminator for enforcing the association between the generated sentence and the image which is the advantage in Jain et al. [40].

III. Design of Autonomous Social Robot

A. System Architecture

The proposed autonomous system of the robot is shown in Fig. 1. The autonomous system interacts with the environment, and the robot will get feedbacks and the sensing information back. We model the internal needs of robot as drives which generates the intensity of motivations with external stimuli sensed by the robot. Since the homeostasis requires to balance the drives, a decision making system is utilized to bound the intensity of motivation. The intensity of motivation is modeled as the state of the decision making system, which decides what actions the robot should take. After the chosen action is executed, we can get the feedback from the user to tune the preference of the user. What's more, the mechanism of moods is built up by the value of drives and stimuli, and it will influence the style of chatting. In the decision making part, deep reinforcement learning algorithm is applied to choose the robot actions, such as charging, resting, finding people, singing or dancing, asking question, chatting, and standing by.

Another main focused point in this work is the module of asking question. For satisfying the need of curiosity, we utilize the deep learning based Visual Question Generation (VQG) for asking question, and combine the variational auto-encoder with Gumbel-Softmax and adversarial training to make the VQG generate multiple different questions.

In addition, the dialogue system of social chat is also made for a more interesting interaction between robot and human, and the system flow is shown in Fig. 2. First, what the human says will be transferred into text by speech-to-text module, and an interactive dialogue management module is going to manage the system how it should respond to the user by QA Bot or by Seq2Seq chit-chat module. As we mentioned previously, the mechanism of moods is a gate for controlling the response being positive or negative style, and the text style translation module will transfer the style of the sentence from the chit-chat module. Eventually, the generated sentence from QA Bot or from Text style translation will be turned into speech to be spoken out by the robot through the text-to-speech module. Due to the reason that providing more information for users, we will show some information on the touch pad of our robot With QA Bot.

In the rest of this chapter, the homeostatic model and decision making system will be described in Section B firstly. Then, we will describe how the dialogue system and the deep learning based text style translation module work in Section C. Finally, the deep learning based visual question generation is mentioned in the last Section D.

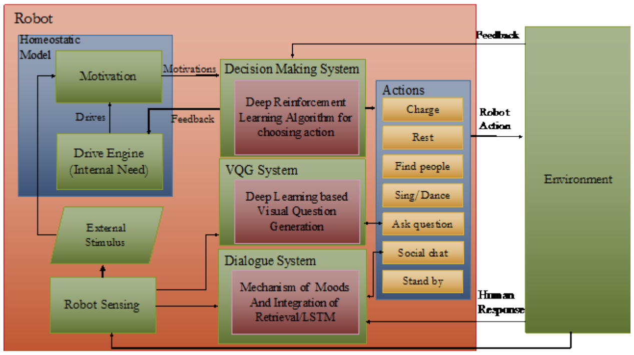


Figure 1. The architecture of the whole system.

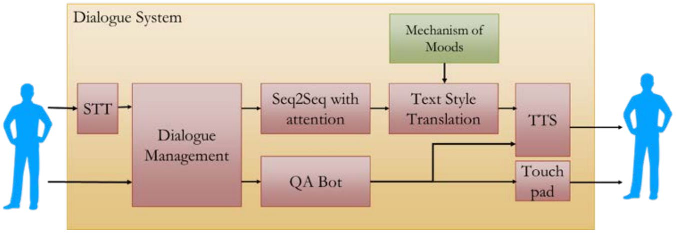


Figure 2. The system architecture of Dialogue System.

B. Homeostatic model and decision making system

In this works, the internal needs of the robot are modeled as drives. Inspired by Maslow's hierarchy of needs, we re-define internal needs in our work for making the robot act like a human as in Table 1. We define 6 kinds of need, namely, Need of Energy (NEner), Need of Safety and Rest (NSaR), Need of Belonging (NBel), Need of Esteem (NEst), Need of Self-Actualization (NSA), and Need of Curiosity (NCur). NEner represents that robot relies on energy to maintain working condition. Because the robot will have some safety issue and overheat issue due to robot motions, NSaR is needed as well by the robot. Just as human needs the company of friends or family, therefore, NBel means the need of the robot for its belonging to a society with human friend. Notice that NEst is the need for the sense of the achievement, which means to offer the services like singing or dancing, because providing service to humans is considered as a main functions of the social companion robot. Also, as a social robot, one crucially important skill is to chat, which refers to the need, NSA. Lastly, behaving like a human, curiosity is a very important characteristic, so NCur is defined as such need of the robot.

Drives are internal parameter of the robot in the form of real numbers, and the value of it indicates the degree of dissatisfaction of the certain need. The set of the drives used in the system is denoted as D. Each drive $d^i \in D$ is normalized to the range [0,1], and smaller value represents that the need is more satisfied. In contrast, the larger value means the need is more severe.

The stimuli are defined as the external conditions that are sensed by the sensors of the robot. There are three kinds of

Table I
Proposed Internal Needs for Social Robot

Maslow's hierarchy of needs	Ours
Need of physiology	Need of Energy (NEner)
Need of safety	Need of Safety and Rest (NSaR)
Need of love and belonging	Need of Belonging (NBel)
Need of esteem	Need of Esteem (NEst)
Need of self-actualization	Need of Self-Actualization (NSA)
	Need of Curiosity (NCur)

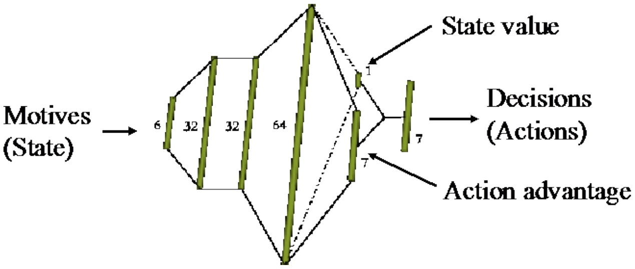


Figure 3. The model architecture of our DDQN.

stimuli, such as Stimulus of Overheat (SOver), Stimulus of Human (SHum), and Stimulus of Encouragement (SEnc). SOver is designed to handle the situation of the overheat of the motors in each joint, in result of having the movement frequently increasing the raising of temperature. It can motivate the robot to know when to take a rest for cooling down the motors. SHum means that whether there is a human in front of the robot or not, whereby the robot will know if it needs to interact with the human currently and also affect the belonging to human for robot. SEnc represents that whether the human give some encouragements to the robot when the robot interacts with the human. While the robot gets the encouragement from the human, the satisfaction of the certain need will continue for a while, which is similar like the feeling of human.

The set of stimuli is denoted as ST, and each stimulus $st^i \in ST$ is defined as 0 or 1 as in the equation (1).

$$\mathrm{st}^i = \begin{cases} 1, & \textit{if the sensor is activated} \\ 0, & \textit{otherwise} \end{cases} \tag{1}$$

Motivations can be considered as the goals for compensating the internal needs of the robot. The intensity of Motivations is generated by the linear combination of value of drives and several external stimuli, and the relation is shown in Table 2. The set of the motivations is denoted as M, and each motivation is denoted as m^i. The relation between drives and motivations is one-to-one, which means that for each drive in the system, the corresponding motivations exists.

Then, the 6 dimensional vector of intensities of motivation will be modeled as the state of our decision making system. Based on the state, we utilize the deep reinforcement learning algorithm, which is Dueling Deep Q-Network (DDQN) [42], to decide the action with the reward function. The reason for choosing a value based reinforcement learning is that policy based reinforcement learning and actor-critic based reinforcement learning are easily reach to optimal goal and stuck there. However, the value based method can take the advantage for reaching the global optimal more easily. We hope that the robot can adapt to and kinds of environment, so we set the initial situation randomly while training. Therefore, to our best knowledge and empirical testing, both policy based method and actor-critic based method cannot work better than value based method like DDQN. The model architecture in this work is shown in Fig. 3.

TABLE II
PROPOSED MOTIVATIONS FOR SOCIAL ROBOT

Drive/Stimulus	Motivation
NEner/<N/A>	Motivation of Survival
NSaR/ SOver	Motivation of Rest
NBel/ SHum, SEnc	Motivation of Relationship
Nest/ SHum, SEnc	Motivation of Achievement
NSA/ SHum, SEnc	Motivation of Social Interaction
NCur/ SHum, SEnc	Motivation of Asking

The reward function is to encourage the right action at the right moment, and the goal here is to balance the value of the drives, which will be decreased after execution of the action, and influence the intensity of the motivation. The reward will mainly be defined in two ways. First, if the action α^i is not standing by, the reward will be calculated by the difference between the current m^i and $m^i_{desired}$. For example, if the desired range of m^i is [0.4, 0.6] with $m^i = 0.5$, then you will get a positive reward κ×(1+0.5-0.6)=0.9κ, which is less than 1. Secondly, if the executed action is “standing by,” the value of m^i in the desired range will be calculated as $\Sigma_i r^i$. The reward will be determined by whether $\Sigma_i r^i$ becomes greater than the threshold or not. For example, if the robot chooses the action “standing by” as in the previous example, it has higher possibility to get a higher reward. The detail of the reward function is shown in equation (2) and (3).

Moreover, the personality like whether the robot is more interested at interacting with the human can be designed by tuning the desired range of the motivation. For example, if the desired range is smaller, the intensity of the motivation will easily exceed the range. Then, the robot will do the specific action frequently to keep the motivation in the desired range. For adapting to different users’ preference, there will be an external reward to deal with the feedback from the users.

$$\begin{aligned} &\text{If the executed action } a^i \text{ was not standing by, then:} \\ &r_{internal} = \begin{cases} \kappa \times \left(1 + \mathrm{m}^i - \mathrm{m}^i_{\mathrm{upper}}\right), \text{ if } \mathrm{m}^i \text{ in DR or } \mathrm{m}^i_{prev} > \mathrm{m}^i_{\mathrm{upper}} \\ \lambda \times \left(1 - \left(\mathrm{m}^i - \mathrm{m}^i_{\mathrm{upper}}\right)\right), \text{ otherwise} \end{cases} \\ &\text{Otherwise} \\ &r_{internal} = \begin{cases} 1, \text{ if } \sum_i r^i > v \\ -1, \text{ otherwise} \end{cases} \\ &r^i = \begin{cases} 1, \text{ if } \mathrm{m}^i \text{ in desired range} \\ 0, \text{ if } \mathrm{m}^i \text{ not in desired range} \end{cases} \end{aligned} \tag{2}$$

$r_{internal}$: internal reward, $r_{external}$: external reward
r^i : the reward correspond to the motivation m^i
κ : the weighted parameter for positive reward, $0 < \kappa < 1$
λ : the weighted parameter for negative reward, $\lambda < 0$
$\mathrm{m}^i_{\mathrm{upper}}$: the desired upper bound intensity of motivation m^i
m^i_{prev} : the intensity of motivation before action was done
v : the threshold for determining the reward of standing by action, and it is an integer. DR: desired range of m^i

$$r_{external} = \begin{cases} 1, & \textit{positive feedback} \\ -1, & \textit{negative feedback} \\ 0, & \textit{otherwise} \end{cases} \tag{3}$$

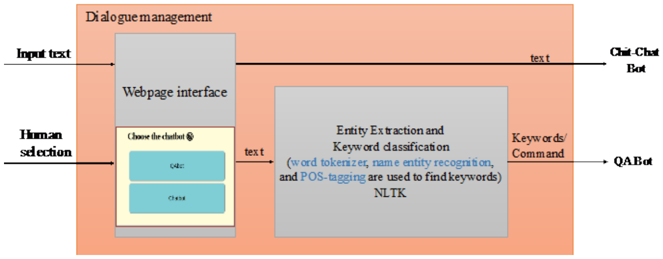

Dialogue management
Input text
Webpage interface
text
Chit-Chat Bot
Human selection
Choose the chatbot
QAbot
Chatbot
text
Entity Extraction and Keyword classification (word tokenizer, name entity recognition, and POS-tagging are used to find keywords) NLTK
Keywords/ Command
QABot

Figure 4. The flow of dialogue management.

C. *Dialogue System*

Dialogue system is utilized to handle the action of social chatting. In the human-robot interaction via chatting, we will utilize the speech to text (STT) module to transfer the speech into text, and use dialogue management to choose the kind of dialogue preferred by the human. If chit-chat is chosen by the human, a sequence to sequence chat-bot, to which the text style transfer module is connected, is applied. If the Question-Answering Bot (QA Bot) is operating, some specific kinds of questions can be answered. After answering is done from the text style transfer module or QA Bot module, the answers will be spoken out by the text to speech (TTS) module of the robot. In the dialogue management, a webpage is made for human to choose different kinds of chatting, and it is shown on the touch pad mounted on the robot. The user just needs to touch the button on the pad to choose. After the preferred button is pressed, the text will be passed directly to “the sequence to sequence with attention chit-chat module” or passed through the “entity extraction and keyword classification module” to select different QA functions.

There are 8 types of QA functions, such as information retrieval, weather, time, news, related topics, google map, playing music, and taking photo. We will first try to tell the category by certain words ζ, and utilize the Natural Language Toolkit (NLTK), like word tokenizer, part-of-speech (POS) tagging, and name entity recognition, to get useful message for answering. In other words, the function will extract the keyword query for the QA Bot. After knowing the category of the question, the label of name entity of each word will be checked first. If the word has the label, the word will be added to the keyword query list. Then, check each POS tagging for the word that is without label of name entity. The words with useful POS tagging will also be added into the keyword query list. Finally, one utilizes the keywords for the QA Bot. The flow of dialogue management is shown in Fig. 4.

The deep learning based chit-chat bot module is applied here, and it is a sequence to sequence model with attention mechanism [43]. This model is an auto-encoder model for encoding the input text into the latent space, and the attention mechanism for focusing on the input tokens is utilized while generating the output sentence. The optimization method is the basic cross-entropy loss to calculate the generated tokens in the discrete sampling way.

The output sentence of the chit-chat bot and the output mood from the mechanism of the moods will be the input of the text style translation module, which is a function to translate the sentence from mood to another. In this work, there are three main factors for influencing the moods of the robot, and they are the value of the drives, the emotion of the human in interaction, and the encouragement from the human. Note that an unsatisfied robot’s need will make the robot become more negative, which is just similar to the human case where one feels anxious and negative when he/she suffers a severe need, like, hunger. In contrast, while the needs are satisfied, the robot act in a more positive way. In addition, if the human has negative emotion, the robot can speak in a more positive way to cheer him/her up. If the human has positive emotion, the robot however can speak in negative way to make joke of the human. On the other hand, the mood of the robot can also be affected if the human encourages the robot; i.e., if there is encouragement, the robot intends to act in a positive way. The detail of the mechanism is shown in the following via equation (4).

$$\begin{aligned} \mathrm{v}_{mood} &= \sum_i w_i \cdot d^i - \mu st_{encouragment} + \varepsilon\beta_{emotion} \\ \mathrm{Mood} &= \begin{cases} Positive,\ if\ \ \mathrm{v}_{mood} \le \eta \\ Negative,\ if\ \ \mathrm{v}_{mood} > \eta \end{cases} \end{aligned} \tag{4}$$

$\sum_i w_i = 1,\ w_i$: weight parameter for each value of druve

μ, ε: weighted factor of encouragement and emotion of human

d^i : the value of drive, $st_{encouragement}$: the stimulus of encouragement

$\beta_{emotion}$: the observation of human's emotion

η : the threshold of determining the mood of the robot

Our text style translation is considered as an unsupervised learning problem, because there are few parallel corpora to use. The architecture of the module is shown in Fig. 5, and we will introduce the training process firstly. There are sentence data, x_1 and x_2, from different domains, $\mathcal{X}_1$ and $\mathcal{X}_2$, respectively, and their respective labels for each style, say, δ_1 and δ_2, will be encoded into the latent space as y_1 and y_2, respectively, by E_y. Then, the encoder E_z will encode both x_1 with y_1 as initial hidden state and x_2 with y_2 as initial hidden state into the same latent space as z_1 and z_2, respectively, where z is the latent variable. For generating the data from $\mathcal{X}_2$ to $\mathcal{X}_1$, we input the concatenation vector of encoded z and the style y_1 as the initial hidden state in the generator *G*, and the generated sequence of hidden states in Long Short-Term Memory (LSTM) cell is denoted as $\hat{h}_i^2$. In contrast, for generating the data from $\mathcal{X}_1$ and $\mathcal{X}_2$, we input the concatenation vector of encoded z and the style

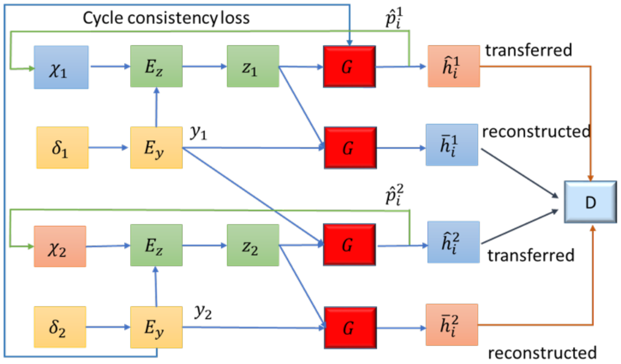


Figure 5. The architecture of text style translation.

y$_2$ as initial hidden state in the generator *G*, and the generated sequence of hidden states in LSTM cell is denoted as $\hat{h}_i^1$. If the style representation is given as the same style of source sentence, the generated sequence of hidden state will be denoted as the reconstructed one, like $\overline{h}_i^1$ and $\overline{h}_i^2$. The model will be pretrained to speed up the convergence as in Ramachandran *et al.* [44]. The model architecture of E$_z$ is a two layer LSTM; *G* is a one layer LSTM, and E$_y$ is one layer of fully connected layer. The discriminator D is the same architecture as TextCNN [45].

In order to force the encoded latent representations in the same latent space and the generated sentence sharing the same data distribution with desired style of data domain, we utilize N+1 discriminator for the cross-alignment training [26] with the sequence of hidden states, such as $\hat{h}_i^1$ with $\overline{h}_i^2$ and $\hat{h}_i^2$ with $\overline{h}_i^1$. There are three classes of the N+1 discriminator, which are reconstructed sequence with positive style, reconstructed sequence with negative style, and transferred sentence with positive/negative style.

Also, while generating the style representation y, the auto-encoder is applied to make y become more meaningful information. The cycle consistency loss is here to preserve more content from the source sentence, and therefore after the style of generated sentence of tokens, $\hat{p}_i^1$ or $\hat{p}_i^2$, is transferred, the generated sentence should be possible to transfer back to the original source sentence.

While testing, the style representation of the current mood of robot will be inputted to the generator, and the opposite style label will be inputted to the style encoder. Then we can get the transferred sentence to be the output sentence for robot to speak out.

D. *Visual Question Generation*

The architecture of the VQG in this work is shown in Fig. 5. In training process, an image encoder E_I encode the information of the image, and the encoder E$_z$ will encode both encoded image features and the sentence, x∈$\mathcal{X}$, into latent space with the latent variable z by Categorical Variational Auto-Encoder with Gumbel-Softmax techniques [28]. It will encode the last hidden state information in encoder with random noise g sampled from the Gumbel distribution G$_{um}$ by the reparameterization trick. The generator G will generate the sentence with the initial state as the concatenation of the latent representation z and encoded image feature. The same techniques of Gumbel-Softmax of continuous relaxation for discrete sampling process and the Professor Forcing for aligning the sentence population is applied here, and is used to align two sequences of hidden states by the discriminator D$_1$ for adversarial training. One of them is that the inputs are the ground truth tokens, and another is that the inputs are the previous generated logits with continuous relaxation. In addition, because some generated sentences do not match to the image in other previous works, the adversarial training is utilized for improving the relatedness between the generated sentence and the image. The discriminator D$_2$ will try to classify the difference between the generated sentence with inputs of previous logits with encoded image feature and the ground truth sentence with the encoded image feature.

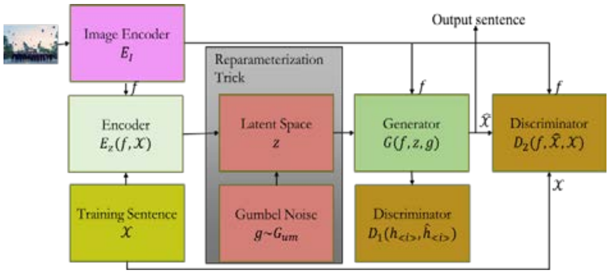


Figure 6. The architecture of Visual Question Generation.

Figure 7. Pepper Robot.

The model architecture of E_I is a VGG16 model [46] to the first fully connected layer with 4096 neurons which is pre-trained on the ImageNet and the parameter will be fixed while training, and it will connected to a fully connected layer with 512 neurons to output the image feature. Both E$_z$ and G are one layer LSTM, and D$_1$ is also the same architecture of TextCNN. D$_2$ is a CNN-LSTM architecture, which inputs the tokens into CNN-LSTM layer and then do the dot product operation with the last hidden state of LSTM and image feature vector.

IV. EXPERIMENT

The experiment of our proposed system will also be break down to three sections, and one is for homeostatic model and decision making system; another is for text style translation; the other is for visual question generation. The robot we used in the experiment is Pepper, and it is shown in Fig. 6.

A. *Homeostatic Model and Decision Making System*

Due to the algorithm of decision making system, the experiment is done in the simulation, and it is evaluated with two metrics. The first one is the reward function of the algorithm for evaluating the result of the deep reinforcement learning algorithm, and the second one is the Robot Secure Rate (RSR) [16][17] for evaluating that whether the robot is secure or not. The reward can approach 650 after training 4000 epochs, and the Figure 7 shows the rewards, which is averaged for every 30 epochs, in the learning process. While testing for 100 epochs, the average reward is 770.19. The reason that reward in testing is higher than in the training is due to the epsilon greedy strategy in training, which chooses random action for exploration.

In RSR index, when a drive exceeds 0.9, it is considered as insecure for our social robot. Thus, the metric representing the time ratio of robot being in secure robot is defined as in equation (5). RSR=0 means the social robot is always insecure, and RSR=1 means the social robot is always secure. The

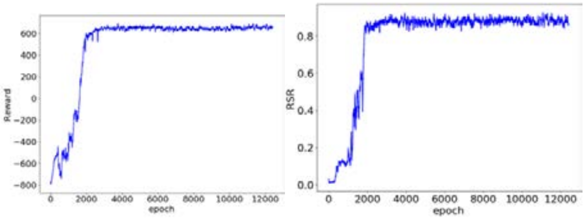


Figure 8. The reward and RSR changing in the training process.

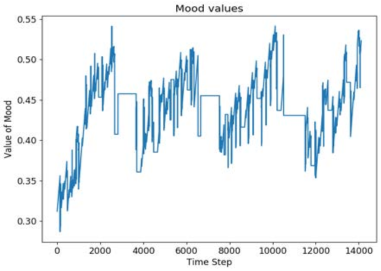


Figure 9. The mood changing in one testing epoch.

performance is shown in Table. , and it has a better performance compared with [16][17] which consider only the dominant motivation.

$$\text{RSR}=\frac{1}{\text{N}}\sum_{i=1}^{N} se(t)$$
$$se(t)=\begin{cases}1, \ if \ \ \forall d_t^i \in D : d_t^i < 0.9\\ 0, \ otherwise\end{cases} \tag{5}$$

where N is the total time step, *se*(*t*) is an indicator for showing if the robot is secure at t time step

The changing of mood value is shown in Figure 8, and if the threshold is defined as 0.44, the mood transition is stable for each time period.

B. *Text Style Translation*

The experiment is evaluated on dataset which is provided by Shen et al. [26], and it is the Yelp dataset. It is a restaurant reviews dataset, which utilizes the user ratings to classify the positive reviews and the negative reviews, and the sentences longer than 15 words are eliminated. Finally, there are 350K positive sentences and 250K negative sentences, and the words appear less than 5 times will be unknown tokens. 270K positive sentences and 180K negative sentences are for training, and 76K positive sentences and 50K negative sentence for testing, while the rest of them are for validation. In order to evaluate the transferred sentences, a model-based evaluation is used. A pre-trained sentiment classifier, which is TextCNN, evaluates whether the sentence has the correct style, and it has 97.47% accuracy on the testing data. As a baseline, we compare against the model with Hu et al. [25] and Shen et al. [26], which is also provided in Shen et al. [26]. The result is shown in the Table 4. While we are testing the code from Shen's github, they revised the program at March and got the score of 83.94%. Our result has a better accuracy as compared with them. In addition, our method without style auto-encoder will get a lower accuracy, which represents that style auto-encoder really help to improve the accuracy by extracting and preserving better style latent representation.

TABLE. III
THE PERFORMANCE OF RSR INDEX.

Work	**RSR index**
Lin et al. [16]	0.9
Yang et al. [17]	0.98
Our proposed one	0.9994

TABLE IV
THE RESULT OF TEXT STYLE TRANSLATION

Method	**Accuracy**
Hu *et al.* (2017)[25]	83.5
Shen *et al.* (2017)[26]: Cross-aligned auto-encoder	78.4
Shen *et al.* (2018 github code)	83.94
Ours (W/O style auto-encoder)	82.05
Ours (W/ style auto-encoder)	84.65

Because of not sharing the same dialogue dataset yet, it will have some difficulties while transferring the sentence for different styles. It will be our future work to find a shared dataset to train both models. We demonstrate some current examples in Table 5.

C. *Figures and Tables*

For the experiment, we combine the VQG dataset from Microsoft [47] and the VQA dataset [48] together, because the questions in VQG dataset are more natural than VQA dataset. However, the sentences in VQG dataset are not enough for the training, so VQA dataset is used. There are approximately 120K images and 3~5 sentences for each image in total. Evaluations of the generated sentences with the image are human evaluation, and there are three main evaluation metrics, such as "Diverse Meaning", "Relatedness with the Image", and "Relatedness with the Scene". The reason that we use the human evaluation is due to the goal of generating diverse sentence for the image. There is no specific ground truth sentence for each image, and we should consider the meaning of the sentence, because it is weird for robot to ask similar sentences just by changing some prepositions in the sentence.

TABLE V
SAMPLE OF TEXT STYLE TRANSLATION

User speak/ Chit-Chat model	**Transferred to negative**	**Transferred to positive**
How was your day today? / Pretty good, how about you?	Pretty bad, but you can get?	Pretty good, you can like!
Do you feel good? /No, I feel very sad today.	No, I really felt disappointed with this place.	Totally a very nice experience with friends.
How was your lunch? / It is very delicious!	It is not very delicious!	It is very delicious!
How was your breakfast? / It is gross.	It is gross.	It is delicious.

- **Diverse Meaning (DM)**

Check if the meaning of the generated sentences is repetitive or not. For example, “how many people are there?” and “how many laptops are there?” are different meaning, while “how many people are there?” and “how many people are here?” are the same meaning.

$$\mathrm{DM}(S)=\frac{1}{\mathrm{N}}\sum_{i=1}^{N}dm\left(s^{i}\right),\ \ s^{i}\in S$$
$$dm\left(s^{i}\right)=\begin{cases}1, \ if\ \ the\ meaning\ if\ \ s^{i}\ \ different\ \ ftom\ others\\ 0,\ otherwise\end{cases} \tag{6}$$

where s^i is the generated sentence in the set of generated sentences *S*, and *N* is the number of sentences in the set.

- **Relatedness with the Image (RI)**

Check if the generated sentence describes the image correctly or not. For example, the model sometimes generates the sentence: “is the dog sleeping?” for a cat in the image, and it will be counted as not related to the image.

$$RI(S)=\frac{1}{\mathrm{N}}\sum_{i=1}^{N}ri\left(s^{i}\right),\ \ s^{i}\in S$$
$$ri\left(s^{i}\right)=\begin{cases}1,\ if\ \ s^{i}\ \ describes\ the\ image\ correctly\\ 0,\ otherwise\end{cases} \tag{7}$$

where s^i is the generated sentence in the set of generated sentences *S*, and *N* is the number of sentences in the set.

- **Relatedness with the Scene (RS)**

Check if the generated sentence is related to the scene of the image and other sentences. For example, model generates the sentence: “is there pizza on the table?” for an image of hotpot restaurant. Although there is not pizza in the image, this sentence is still around the scene of food, and it will be considered that it is related to the topic.

$$RS(S)=\frac{1}{\mathrm{N}}\sum_{i=1}^{N}ri\left(s^{i}\right),\ \ s^{i}\in S$$
$$rs\left(s^{i}\right)=\begin{cases}1,\ if\ \ s^{i}\ \ is\ related\ \ to\ the\ topic\\ 0,\ otherwise\end{cases} \tag{8}$$

where s^i is the generated sentence in the set of generated sentences *S*, and *N* is the number of sentences in the set.

We build our testing dataset by finding the images from internet, and some images are more common in daily life for robot to observe in real world. There are 50 testing images, and we use 100 sampling noises to generate the sentences. Finally, the model can generate 15.7 sentences in average, which means 785 sentences in total. The score of DM is 0.879, and it means that there are 87.9% sentences having different meaning for each image. The score of RI is 0.917 representing that 91.7 % of generated sentences can accurately describe the image. The score of RS is 0.996, and it means that almost all the generated sentences are related to the scene of the image. One example is shown in Fig. 10.

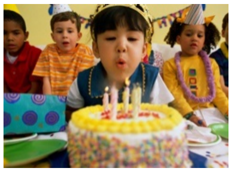
V. CONCLUSION

We proposed a novel autonomous system based on the homeostatic theory with inspiration from Maslow’s hierarchy of needs for a social companion robot acting in a human manner, and built up the mechanism of moods for it. A robot with moods can improve the human-robot interaction, and moods are also going to be used in the dialogue system. As a social companion robot, dialogue system is indispensable, and there are two main functions. First, the chit-chat bot with the text style translation for transferring the positive or negative style, which is decided by the mechanism of the moods. Second, the QA bot for answering general questions is also built up. In addition, in order to make robot behaves in more humanlike way, we model the internal need of curiosity that is an important characteristic of human. The visual question generation module with visual perception system of robot are proposed to handle the curiosity need and to take initiative in the interaction.

Dueling Deep Q Network (Dueling DQN) is utilized for the decision making part of the autonomous system, and it can reach a more global optimal goal by modeling the continuous state, which is the intensity of motivations. For modeling different personality for the robot, the desired range of the intensity of motivation can be changed to make robot perform different dedication to interact with the human.

In the dialogue system, the chit-chat bot is a Seq2Seq with attention model, and the data driven based text style translation is proposed. In text style translation module, the pre-trained language model is used to speed up the convergence, and combine the cross-alignment training for aligning the sentence population of two styles. Moreover, N+1 discriminator is used for decreasing the parameters of the model and a style auto-encoder is used to extract more meaningful information of style. To preserve the content of the sentence, cycle-consistency loss is also used for updating. In the QA bot, there are functions, such as information retrieval type, asking weather, asking time, asking news, asking related topics, asking for google map, asking for playing music, asking for taking photo.

Deep learning based visual question generation deploys the categorical variational auto-encoder with Gumbel-Softmax for encoding the latent representation, and utilize the



'what is on the table ?', 'are there any children in the picture ?', 'what color is the tablecloth ?', 'how many people are in the picture ?', 'how many candles are on the cake ?', 'are the candles lit ?', 'are these children ?', 'is the woman wearing glasses ?', 'is this a birthday cake ?', 'what color is the cake ?', 'what are the people eating ?', 'are these people eating ?', 'what is the girl eating ?', 'what is the woman holding ?', 'are the children happy ?', 'what occasion is being celebrated ?'

Figure 10. The example of Visual question generation.

Professor-Forcing algorithm and continuous relaxation for discrete sampling process to avoid bias exposure. We also used the adversarial training to match sentences with the image better.

In the experiment of autonomous system, the Robot Secure Rate (RSR) can reach 99.94% in the testing, which means the robot can satisfy the internal needs well. For the text style translation module, we used a pre-trained sentiment classifier to determine the performance, and it has 84.65% accuracy to successfully transfer the style. The evaluation metrics of diverse meaning, relatedness of image, and relatedness of the topic are used for the visual question generation module, and it is done by the human evaluation. We can get 15.7 sentences for a single image in average, and get 87.9% sentences with different meaning. There are 91.7% generated sentences described the image correctly, and 99.6% of generated sentences are related to the same topic of the image.

There are future works for each part in this work. In autonomous system, changing the personality of the robot by tuning desired range of intensity of motivation should be in a more automated way. In the dialogue system, in order to combine the chit-chat bot and text style translation module, we should find a better dialogue dataset for training both of them. Otherwise, the text style translation is difficult to transfer the style of sentence outputted from the chit-chat bot. In the visual question generation, it can be developed as a visual understanding module by getting the answer back from the human. Most important of all, we will target at the companion ability in our future work, such as focusing on a specific human who needs companion and maximizing the usage of our mood mechanism. In addition, we should also try to develop a better evaluation for evaluating the users' feeling and the companion ability of the robot.

References

[1] Jibo, Inc., *Jibo Robot – He can't wait to meet you*, May 30, 2018. Accessed on: Sep. 29, 2019. [Online]. Available: https://www.jibo.com/

[2] ASUSTek Computer Inc., *Zenbo*, July 8, 2019. Accessed on: Sep. 29, 2019. [Online]. Available: https://zenbo.asus.com/tw/

[3] SoftBank Robotics, *Pepper the humanoid robot*, Apr. 15, 2018. Accessed on: Sep. 29, 2019. [Online]. Available: https://www.softbankrobotics.com/emea/en/robots/pepper/find-out-more-about-pepper

[4] T. Breuer, G. R. G. Macedo, R. Hartanto, N. Hochgeschwender, D. Holz, F. Hegger, Z. Jin, C. Müller, J. Paulus, M. Reckhaus, "Johnny: An autonomous service robot for domestic environments," Journal of intelligent & robotic systems, vol. 66, no. 1-2, pp. 245–272, 2012.

[5] iRobot Coorp., *irobot: Your partner for a cleaner home*, 2016, accessed: 10-July-2016. [Online]. Available: www.irobot.com

[6] W. B. Cannon, The wisdom of the body. WW Norton & Co, 1932.

[7] Saul McLeod. "Maslow's hierarchy of needs." Simply Psychology 1 (2007).

[8] Rachel Gockley, Jodi Forlizzi, and Reid Simmons. "Interactions with a moody robot." Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction, 2006.

[9] Iolanda Leite, Ginevra Castellano, André Pereira, Carlos Martinho, and Ana Paiva. "Long-term interactions with empathic robots: Evaluating perceived support in children." International Conference on Social Robotics. Springer, Berlin, Heidelberg, 2012.

[10] David Feil-Seifer and Maja J. Mataric. Socially assistive robotics: Ethical issues related to technology. IEEE Robotics and Automation Magazine, 18(1):24–31, 2011

[11] Muhammad Ali, Samir Alili, Matthieu Warnier, and Rachid Alami. An Architecture Supporting Proactive Robot Companion Behavior. In New Frontiers in Human- Robot Interaction at AISB, 2009

[12] R. C. Arkin, M. Fujita, T. Takagi, and R. Hasegawa, "An ethological and emotional basis for human–robot interaction," Robotics and Autonomous Systems, vol. 42, no. 3, pp. 191–201, 2003.

[13] C. L. Breazeal, Designing sociable robots. MIT press, 2004.

[14] SoftBank Robotics, NAO the humanoid robot, Apr. 15, 2018. Accessed on: Oct. 18, 2019. [Online]. Available: https://www.softbankrobotics.com/emea/en/nao

[15] H.-L. Cao, P. G. Esteban, A. De Beir, R. Simut, G. Van de Perre, D. Lefeber, and B. Vanderborght, "Robee: A homeostatic-based social behavior controller for robots in human-robot interaction experiments," in Proceedings of IEEE International Conference on Robotics and Biomimetics (ROBIO), 2014, pp. 516–521.

[16] Ching Lin, Rui-Hong Zhang, Shih-Huan Tseng, and Li-Chen Fu, "A homeostasis based decision making system on human-aware autonomous service robot." International conference on Advanced Robotics and Intelligent Systems (ARIS), 2017.

[17] Chiao-Yu Yang, Ming-Jen Lu, Shih-Huan Tseng, and Li-Chen Fu, "A companion robot for daily care of elders based on homeostasis." 2017 56th Annual Conference of the IEEE Society of Instrument and Control Engineers of Japan (SICE).

[18] Á. Castro-González, M. Malfaz, and M. A. Salichs, "Selection of actions for an autonomous social robot," in Proceedings of International Conference on Social Robotics, 2010, pp. 110–119.

[19] D. Cañamero, "Modeling motivations and emotions as a basis for intelligent behav- ior," in Proceedings of the first International Conference on Autonomous Agents. ACM, 1997, pp. 148–155.

[20] S. P. Gadanho, "Reinforcement learning in autonomous robots: an empirical inves- tigation of the role of emotions," 1999.

[21] S. C. Gadanho and L. Custódio, "Asynchronous learning by emotions and cognition," in Proceedings of the seventh International Conference on Simulation of Adap- tive Behavior on From Animals to Animats. MIT Press, 2002, pp. 224–225.

[22] Jessica Ficler, and Yoav Goldberg. "Controlling linguistic style aspects in neural language generation." arXiv preprint arXiv:1707.02633 (2017).

[23] Mengqiao Han, Ou Wu, and Zhendong Niu. "Unsupervised Automatic Text Style Transfer Using LSTM." National CCF Conference on Natural Language Processing and Chinese Computing. Springer, Cham, 2017.

[24] Jonas Mueller, Tommi Jaakkola, and David Gifford. Sequence to better sequence: continuous revision of combinatorial structures. International Conference on Machine Learning (ICML), 2017.

[25] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. "Toward controlled generation of text." International Conference on Machine Learning. 2017.

[26] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. "Style transfer from non-parallel text by cross-alignment." Advances in Neural Information Processing System, 2017.

[27] Alex M Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. "Professor forcing: A new algorithm for training recurrent networks." In *Advances Neural Information Processing Systems*, pages 4601–4609, 2016.

[28] Eric Jang, Shixiang Gu, and Ben Poole. "Categorical reparameterization with gumbel-softmax." arXiv preprint arXiv:1611.01144, 2016.

[29] Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. "Unsupervised Machine Translation Using Monolingual Corpora Only." International Conference on Learning Representations, 2018.

[30] Yanpeng Zhao, Victoria W. Bi, Deng Cai, Xiaojiang Liu, Kewei Tu, Shuming Shi. "Language Style Transfer from Non-Parallel Text with Arbitrary Styles." International Conference on Learning Representations, 2018

[31] Igor Melnyk, Cicero Nogueira dos Santos, Kahini Wadhawan, Inkit Padhi, and Abhishek Kumar. "Improved Neural Text Attribute Transfer with Non-parallel Data." In Advances in Neural Information Processing Systems Workshop on Learning Disentangled Representations: from Perception to Control, 2017.

[32] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. "Improved techniques for training gans." Advances in Neural Information Processing Systems. 2016.

[33] Somak Aditya, Yezhou Ang, Chitta Baral, Cornelia Fermuller, and Yiannis Aloi- monos. From images to sentences through scene description graphs using reasoning and knowledge. arXiv preprint arXiv:1511.03292, 2015.

[34] A. Gatt and E. Reiter. Simplenlg: A realisation engine for practical applications. In Proceedings of the 12th Eu- ropean Workshop on Natural Language Generation, ENLG ’09, pages 90–93, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[35] Yi-Luen Wu, “Interactive Question-Posing System for Reminiscing about Personal Photos,” M.S. thesis, Intelligent Robot and Automation Lab., National Taiwan Univ., Taiwan, 2017.

[36] Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, Anton van den Hengel. "Image captioning and visual question answering based on attributes and external knowledge." IEEE transactions on pattern analysis and machine intelligence (2017).

[37] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. "Densecap: Fully convolutional localization networks for dense captioning." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[38] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015.

[39] Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. "Towards diverse and natural image descriptions via a conditional gan." IEEE International Conference on Computer Vision (ICCV), 2017.

[40] Unnat Jain, Ziyu Zhang, and Alexander Schwing. "Creativity: Generating diverse questions using variational autoencoders." Computer Vision and Pattern Recognition. Vol. 1. 2017.

[41] Liwei Wang, Alexander Schwing, and Svetlana Lazebnik. "Diverse and Accurate Image Description Using a Variational Auto-Encoder with an Additive Gaussian Encoding Space." Advances in Neural Information Processing Systems. 2017.

[42] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, Nando de Freitas. "Dueling network architectures for deep reinforcement learning." arXiv preprint arXiv:1511.06581 (2015).

[43] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." arXiv preprint arXiv:1409.0473 (2014).

[44] Prajit Ramachandran, Peter J. Liu, Quoc V. Le. "Unsupervised pretraining for sequence to sequence learning." arXiv preprint arXiv:1611.02683 (2016).

[45] Yoon Kim. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882 (2014).

[46] Karen Simonyan, Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition.” arXiv preprint arXiv:1409.1556, 2014.

[47] Microsoft, *Download Visual Question Generation dataset from Official Microsoft Download Center*, Sep. 8, 2016. Accessed on: Sep. 29, 2019. [Online]. Available: https://www.microsoft.com/en-ca/download/details.aspx?id=53670

[48] VirginiaTech and Georgia Tech, *VQA: Visual Question Answering*, Jun 2019. Accessed on: Sep. 29, 2019. [Online]. Available: http://www.visualqa.org

Zong-Ze Wu received the M.S. degree in electrical engineering from National Taiwan University, Taiwan, in 2018. His research interests include intelligent robots, visual question generation and answering.

Yu-Ting Hsiao received the M.S. degree in electrical engineering from National Taiwan University, Taiwan, in 2019. Her research interests include intelligent robots, visual question generation and dialogue system.

Xin-Yi Jian received a MS degree in Communications Engineering from National Chiao Tung University in Taiwan in 2009. She is a PhD student at National Taiwan University, Taiwan since 2019. Her current research work is concerned with intelligent robots and CareBots.

Li-Chen Fu received B.S. degree from National Taiwan University, Taiwan, in 1981, and M.S. and Ph.D. degrees from University of California, Berkeley, U.S.A. in 1985 and 1987, respectively.

Since 1987, he joined Dept. of Electrical Engineering and Dept. of Computer Science and Information Engineering, National Taiwan University (NTU), Taiwan, R.O.C. as a faculty member, and was awarded Lifetime Distinguished Professorship in 2007. Currently, he serves as Director of NTU Center for Artificial Intelligence (AI) and Advanced Robotics as well as Co-director of MOST (Ministry of Science and Technology)/NTU Joint Research Center for AI Technology and All Vista Healthcare. So far, he has received numerous recognitions, including IEEE Fellow (2004) and IFAC Fellow (2017). Internationally, he is serving as Editor-in-Chief of the Asian Journal of Control, and Advisory Committee member of Asian Control Association. His research interests include social robotics, smart home, visual detection and tracking, virtual reality, and control theory & applications.

LMI-based Tracking Control of a Omni-directional Mobile Robot Using T-S Fuzzy Sliding Mode

Gwo-Ruey Yu, *Member, IEEE* and Wen-Yen Chen

Abstract—This paper proposed the path tracking design of a mobile robot based on Takagi-Sugeno (T-S) fuzzy modeling method and sliding mode control. The T-S fuzzy model is utilized to represent the equation of motion of the mobile robot, and the concept of parallel distributed compensation (PDC) is applied to design the T-S fuzzy controller. The stability of the system is guaranteed by linear matrix inequalities (LMIs) from Lyapunov approach. The sliding mode control (SMC) preserves the advantages of quick response and low noise. Therefore, the T-S fuzzy control combine the SMC enable a mobile robot system to possess better robustness.

Index Terms*—*Takagi-Sugeno(T-S) fuzzy modeling, mobile robot, sliding mode control.

I. Introduction

IN order to adapt to different environments, the maneuverability of robots is getting more important. As technology advances, we hope to provide a better platform for mobility and maneuverability so that an omni-directional platform design can be proposed to replace the traditional differential drive moving platform [1]. The omni-directional mobile robot (ODMR) is constructed by several orthogonal-wheels inside the wheel [2]. In recent years, the research and technological development of mobile robots are popular and make wide ranges of applications used in the industrial sectors, hospitals, or family, etc. [3].

The mathematical model of an ODMR system was derived [4]. The dynamic equation of the robot is nonlinear [5]. It is a difficult task to design a nonlinear tracking controller. The LMI-based fuzzy control is a better way to approach and solve the complex problem.

The T-S fuzzy is a particular framework. It can be applied to represent a nonlinear system. The feature of T-S fuzzy model is to express the local dynamics of each fuzzy rule by linear system model [6]. After T-S fuzzy model was built, the PDC offers the design of fuzzy controller which shares the same fuzzy sets with the T-S model in the premise parts [7].

In the traditional approach, the feedback gains of PDC are designed via the LMI conditions. To determine the feedback gains of PDC, Lyapunov theorem is applied to derive a sufficient stability condition which guarantees the closed-loop fuzzy is globally stable. The stability condition is expressed in terms of LMI so that the feedback gains of fuzzy controller could be obtained via convex optimization programming technique.

To upgrade the control performance of mobile robot, the sliding mode control has been combined with T-S fuzzy model in this paper. This approach makes the control system have quick respond and better robustness [8].

II. T-S Fuzzy Controller

The T-S fuzzy model proposed by Takagi and Sugeno is described by IF-THEN rules.

A. T-S Fuzzy Model

The main feature of a T-S fuzzy model is constructed by several rules for controlling a non-linear system. Let p be the number of the fuzzy rules. The i th rule can be defined as

Model Rule i :

IF $z_1(t)$ is M_{i1} AND $\cdots$ AND $z_p(t)$ is M_{ip} ,

$$\text{THEN} \begin{cases} \dot{x}(t) = A_i x(t) + B_i u(t) \\ y(t) = C_i x(t) \end{cases} i = 1, 2, \ldots, p \tag{1}$$

where M_{ip} is the fuzzy set, $z_1(t)$,…., $z_p(t)$ are known variables, $x(t)$ is the state vector, $u(t)$ is the input vector, $A_i \in R^{n\times n}$, $B_i \in R^{n\times m}$, $C_i \in R^{l\times n}$.

The state equation of T-S fuzzy model is described as follows:

$$\begin{aligned} \dot{x}(t) &= \frac{\sum_{i=1}^{p} w_i\left(z(t)\right)\left(A_i x(t) + B_i u(t)\right)}{\sum_{i=1}^{p} w_i\left(z(t)\right)} \\ &= \sum_{i=1}^{p} h_i\left(z(t)\right)\left(A_i x(t) + B_i u(t)\right) \end{aligned} \tag{2}$$

where

$$\begin{cases} \sum_{i=1}^{p} h_i\left(z(t)\right) = 1 \\ h_i\left(z(t)\right) \in \left[0 \quad 1\right] \end{cases} \tag{3}$$

Gwo-Ruey Yu is with Department of Electrical Engineering, National Chung Cheng University, Chia-Yi, Taiwan (e-mail: ieewoyu@ccu.edu.tw).

Wen-Yen Chen is with Department of Electrical Engineering, National Chung Cheng University, Chia-Yi, Taiwan.

Research is supported in part by the Ministry of Science and Technology, Taiwan, R.O.C., under Grant Numbers MOST 107-2221-E-194 -044 -MY2.

B. PDC Fuzzy Controller

The PDC fuzzy controller shares the same fuzzy membership with the T-S fuzzy model.

Model Rule i :

IF $z_1(t)$ is M_{i1} and … and $z_p(t)$ is M_{ip}

$$\text{THEN } u(t)=F_i x(t),\ i=1,2,\ldots,p \tag{4}$$

The final outputs of the fuzzy systems are shown as follows

$$u(t)=\sum_{i=1}^{r} h_i(z(t))F_i x(t) \tag{5}$$

where

$$\begin{cases} \sum_{i=1}^{p} h_i(z(t))=1 \\ h_i(z(t))\in[0\quad 1] \end{cases} \tag{6}$$

C. Decay Rate Controller Design Using Relaxed Stability Analysis

In a continuous-time fuzzy system, if there exists a common positive definite matrix *P*. The condition that $\dot{V}(x(t))\le -2\alpha V(x(t))$ for all trajectories is equivalent to [6]

$$G_{ii}^T P+PG_{ii}+(s-1)Q+2\alpha P\le 0 \tag{7}$$

$$\left(\frac{G_{ij}+G_{ji}}{2}\right)^T P+P\left(\frac{G_{ij}+G_{ji}}{2}\right)-Q+2\alpha P\le 0 \tag{8}$$

where $i<j$, $G_{ii}=A_i-B_iF_i$, $G_{ij}=A_i-B_iF_j$, $G_{ji}=A_j-B_jF_i$, s is the number of trigger rules that $1<s\le r$ and Q is the semi-definite matrix.

III. Design of LMI-Based Fuzzy Controller

A. The Four-Wheel ODMR Model

Fig. 1 shows the structure of four-wheel ODMR. Table I lists the symbols of the four-wheel ODMR. Table II lists the value of the parameter.

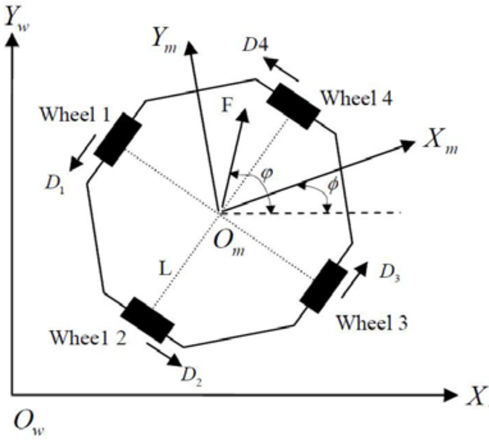


Figure 1. The four-wheel ODMR.

TABLE I
THE SYMBOLS OF THE FOUR-WHEEL ODMR.

Wheel 1	Front left wheel
Wheel 2	Rear right wheel
Wheel 3	Rear left wheel
Wheel 4	Front right wheel
L	The radius of the ODMR
D_i , i=1~4	The traction force of the wheels
ϕ	The angle between X_w and X_m
φ	The azimuth angle of robot

TABLE II
THE VALUE OF THE PARAMETER.

I_w	0.02108 kgm^2
I_v	11.25 kgm^2
r	0.0245 m
c	5.983×10^{-6} kgm^2/s
m	9.4 kg
L	0.178 m

The T-S fuzzy model of this robot is proposed in the literature. The states of the mobile robot $x(t)=\left[\dot{x}(t)\quad \dot{y}(t)\quad \dot{\phi}(t)\right]^T$, the control input variables $u(t)=\left[u_1(t)\quad u_2(t)\quad u_3(t)\right]^T$, and the output variables $y(t)=\left[\dot{x}(t)\quad \dot{y}(t)\quad \dot{\phi}(t)\right]^T$. The dynamic equation of the four-wheel ODMR is

$$\begin{aligned} \dot{x}(t)&=A(x)x(t)+B(x)u(t) \\ y(t)&=Cx(t) \end{aligned} \tag{9}$$

and we have

$$\begin{cases} \dot{x}(t)=\sum_{i=1}^{3} h_i(\dot{\phi}(t)[A_i x(t)+B_i u(t)]) \\ y(t)=\sum_{i=1}^{3} C_i x(t) \end{cases} \tag{10}$$

where

$$A_1=\begin{bmatrix} a_1 & 0 & 0 \\ 0 & a_1 & 0 \\ 0 & 0 & a_3 \end{bmatrix},\ A_2=\begin{bmatrix} a_1 & -a_2\dot{\phi}(t) & 0 \\ a_2\dot{\phi}(t) & a_1 & 0 \\ 0 & 0 & a_3 \end{bmatrix},$$

$$A_2=\begin{bmatrix} a_1 & a_2\dot{\phi}(t) & 0 \\ -a_2\dot{\phi}(t) & a_1 & 0 \\ 0 & 0 & a_3 \end{bmatrix},\ B_1=B_2=B_3=I_{3\times 3}$$

$$C_1=C_2=C_3=I_{3\times 3},\ a_1=-c/\left(I_w+mr^2\right),$$

$$a_2=I_w/\left(I_w+mr^2\right),\ \text{and } a_3=-4cL^2/\left(4I_wL^2+I_vr^2\right).$$

Then gets the state equation as follows

$$\begin{aligned} \ddot{x}_w(t)&=a_1\dot{x}(t)-a_2\dot{\phi}(t)\dot{y}(t)+u_1(t) \\ \ddot{y}_w(t)&=a_2\dot{\phi}(t)\dot{x}(t)+a_1\dot{y}(t)+u_2(t) \\ \ddot{\phi}_w(t)&=a_3\dot{\phi}(t)+u_3(t) \end{aligned} \tag{11}$$

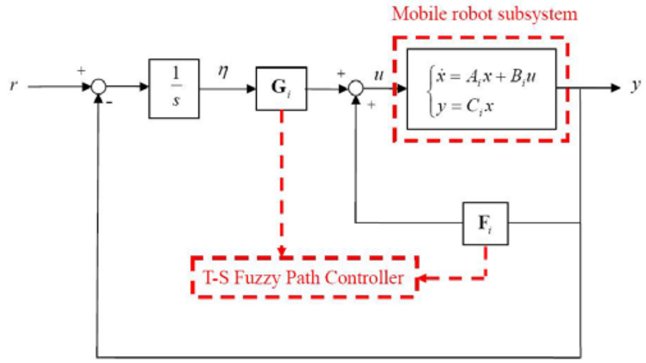


Figure 2. The block diagram of the tracking system.

B. Path Tracking Controller Design

In order to achieve the objective of path tracking, we must be rewrite (14) to the error state equation. First, define the error state variables as follows:

$$\dot{e}(t) = r(t) - y(t) = r(t) - Cx(t) \tag{12}$$

where $r(t) = \begin{bmatrix} \dot{x}_r(t) & \dot{y}_r(t) & \dot{\phi}_r(t) \end{bmatrix}^T$.

Substituting (12) into (11) gets the error state equations as follows:

$$\begin{bmatrix} \ddot{x}_w \\ \ddot{y}_w \\ \ddot{\phi}_w \end{bmatrix} = \begin{bmatrix} a_1 & -a_2\dot{\phi}(t) & 0 \\ a_2\dot{\phi}(t) & a_1 & 0 \\ 0 & 0 & a_3 \end{bmatrix} \begin{bmatrix} \dot{x}_w \\ \dot{y}_w \\ \dot{\phi}_w \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} r_1(t) \\ r_2(t) \\ r_3(t) \end{bmatrix} \tag{13}$$

where $r(t)$ is the input command signal.

To avoid the steady-state error problem, we add a PI controller in the feed-forward loop. Fig. 2 shows the block diagram of the path control system.

From Fig. 2, we can obtain:

$$\begin{aligned} u &= F_i x + G_i \eta \\ \dot{\eta} &= r - y - r - Cx \end{aligned} \tag{14}$$

The augmented state equation is derived as

$$\begin{bmatrix} \dot{x} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ \eta \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r \tag{15}$$

Moreover,

$$\begin{bmatrix} \dot{x}(\infty) \\ \dot{\eta}(\infty) \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x(\infty) \\ \eta(\infty) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(\infty) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(\infty) \tag{16}$$

Let eq. (15) subtract eq. (16), we have

$$\begin{bmatrix} \dot{x} - \dot{x}(\infty) \\ \dot{\eta} - \dot{\eta}(\infty) \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x - x(\infty) \\ \eta - \eta(\infty) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \begin{bmatrix} u - u(\infty) \end{bmatrix} \tag{17}$$

Let $\begin{bmatrix} \dot{x} - \dot{x}(\infty) \\ \dot{\eta} - \dot{\eta}(\infty) \end{bmatrix} = \begin{bmatrix} \dot{x}_e(t) \\ \dot{\eta}_e(t) \end{bmatrix}$, we have

$$\begin{bmatrix} \dot{x}_e(t) \\ \dot{\eta}_e(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x_e(t) \\ \eta_e(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_e(t) \tag{18}$$

Thus, we get the new error state equation:

$$\dot{e}(t) = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} e(t) + \begin{bmatrix} B \\ 0 \end{bmatrix} u_e(t) = A_e e(t) + B_e u_e(t) \tag{19}$$

The total control input is designed as

$$u_e = \begin{bmatrix} F_i & G_i \end{bmatrix} \begin{bmatrix} y_e \\ \eta_e \end{bmatrix} = \begin{bmatrix} F_i & G_i \end{bmatrix} e(t) \tag{20}$$

C. T-S Fuzzy Rules

Fig. 3 shows the membership functions. The T-S fuzzy model is designed by the following rules:

Model Rule 1:

IF $\dot{\phi}(t)$ is about -1

THEN $\dot{e}(t) = \tilde{A}_1 e(t) + \tilde{B}_1 u_e(t)$ and $y(t) = \tilde{C}_1 e(t)$ (21)

Model Rule 2:

IF $\dot{\phi}(t)$ is about 0

THEN $\dot{e}(t) = \tilde{A}_2 e(t) + \tilde{B}_2 u_e(t)$ and $y(t) = \tilde{C}_2 e(t)$ (22)

Model Rule 3:

IF $\dot{\phi}(t)$ is about 1

THEN $\dot{e}(t) = \tilde{A}_3 e(t) + \tilde{B}_3 u_e(t)$ and $y(t) = \tilde{C}_3 e(t)$ (23)

where

$$\tilde{A}_1 = \begin{bmatrix} a_1 & -a_2 & 0 & 0 & 0 & 0 \\ a_2 & a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_3 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}, \quad \tilde{A}_2 = \begin{bmatrix} a_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_3 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix},$$

$$\tilde{A}_3 = \begin{bmatrix} a_1 & a_2 & 0 & 0 & 0 & 0 \\ -a_2 & a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_3 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}, \quad \begin{aligned} &\tilde{B}_1 = \tilde{B}_2 = \tilde{B}_3 = I_{3\times 3} \\ &\tilde{C}_1 = \tilde{C}_2 = \tilde{C}_3 = \begin{bmatrix} C & 0 \end{bmatrix} \end{aligned}$$

Now, we can define the PDC fuzzy controller and the new T-S fuzzy system as follows

Model Rule 1:

IF $\dot{\phi}(t)$ is about -1

THEN $u_e(t) = F_1 x(t) + G_1 \eta(t)$ (24)

Model Rule 2:

IF $\dot{\phi}(t)$ is about 0

THEN $u_e(t) = F_2 x(t) + G_2 \eta(t)$ (25)

Model Rule 3:

IF $\dot{\phi}(t)$ is about 1

THEN $u_e(t) = F_3 x(t) + G_3 \eta(t)$ (26)

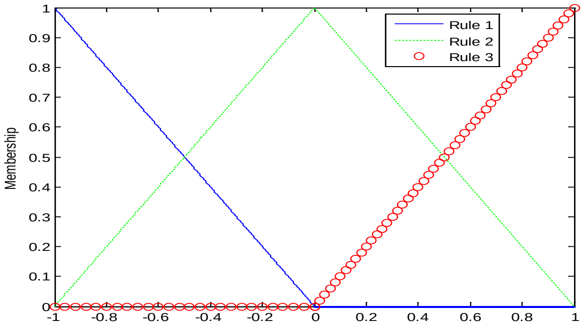


Figure 3. Fuzzy membership function.

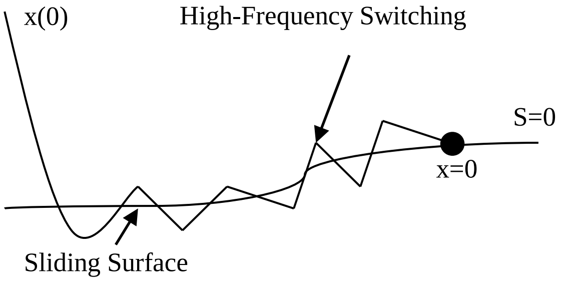


Figure 4. Sliding mode control

IV. Sliding Mode Control

The sliding controller design provides a systematic approach to the problem of sustaining stability and accordant performance for system with model uncertainty. It has been successfully applied to robot arms, electric motors, and fly vehicles in recent years. Thus, sliding mode control has been studied popular in many fields due to the great advantage in quick response and better robustness.

Sliding mode control must work at least two subsystems. Using the approaching condition, the system states approach into the sliding surface. Using the sliding condition, the system states will achieve the control object on the sliding surface. Fig. 4 shows the state variation under sliding mode control.

In general, there are two steps for designing sliding mode controller.

Step 1. Choose a suitable sliding surface $s(x)$.

Step 2. Design a variable structure control u which will drive the system state to the sliding surface in finite time.

Generally, the sliding surface can be set as

$$s(x) = c_1x_1 + c_2x_2 + \cdots + c_{n-1}x_{n-1} + c_nx_n \tag{27}$$

For a system $\dot{x} = f(x) + g(x)u$, the sliding mode control law can be chosen as

$$u = -\left(\frac{\partial s}{\partial x}g(x)\right)^{-1}\frac{\partial s}{\partial x}f(x) - \left(\frac{\partial s}{\partial x}g(x)\right)^{-1}k\cdot\operatorname{sgn}(s), k > 0 \tag{28}$$

In this paper, the authors combine the T-S fuzzy model and sliding mode control to treat with the path control of the mobile robot. Let the error state equation (13) to be rewritten as

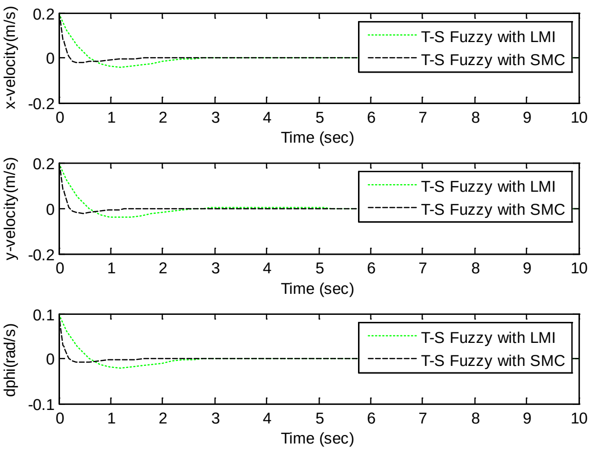


Figure 5. The velocity responses of the mobile robot

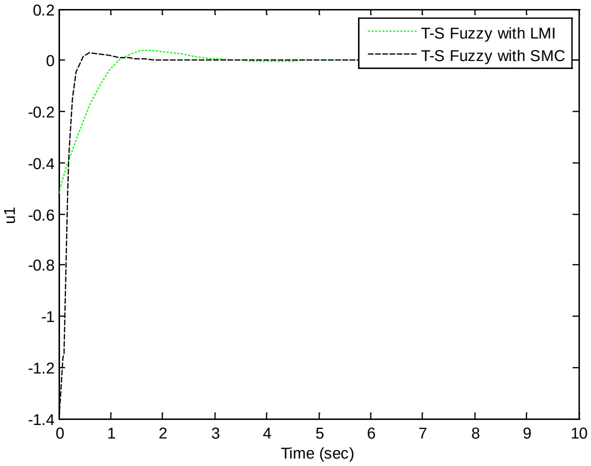


Figure 6. The variation of u_1

$$\begin{aligned}\dot{\xi}_1 &= \dot{\xi}_2\\ \dot{\xi}_2 &= Q(\xi) + Z(\xi)(u+d)\end{aligned} \tag{29}$$

where $Q(\xi) = \begin{pmatrix} a_1\dot{x}(t) - a_2\dot{\phi}(t)\dot{y}(t) \\ a_2\dot{\phi}(t)\dot{y}(t) + a_1\dot{y}(t) \\ a_3\dot{\phi}(t) \end{pmatrix}$, $Z(\xi) = I$ and d is the disturbance signal. The T-S fuzzy sliding control law will be

$$u = u^{eq} + u^{re} \tag{30}$$

where

$u^{eq} = \left(\sum_{i=1}^{p} w_i(x)B_i(x)\right)^{-1}\cdot\left[-\sum_{i=1}^{p} w_i(x)A_i(x)x + \ddot{x}_d - K\dot{e}\right]$,

$u^{re} = -\eta\left(\sum_{i=1}^{p} w_i(x)B_i(x)\right)^{-1}\cdot sat(s)$, s is the sliding surface, and η is positive.

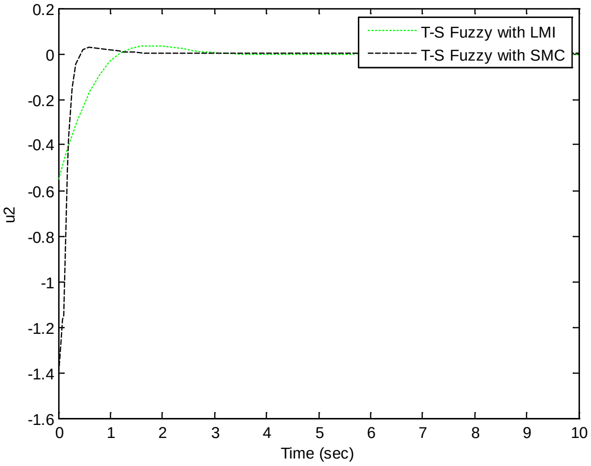


Figure 7. The variation of u_2

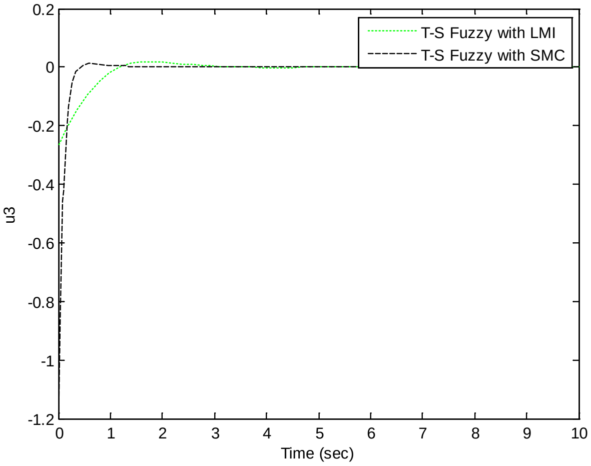


Figure 8. The variation of u_3

V. Computer Simulation

A. Path Regulation

To show the effects of the T-S fuzzy sliding control, the conventional T-S fuzzy control is applied to the mobile robot for comparison. Let the reference commands be step input signal. The initial conditions of x , y , ϕ are 0.2, 0.2 and 0.1. Fig. 5 shows the responses of x-velocity, y-velocity and angle velocity, respectively. Obviously, the T-S fuzzy sliding controller is faster than the conventional T-S fuzzy controller. Fig. 6~8 shows the responses of control input u_1 , u_2 and u_3 , respectively. Obviously, the energy of the T-S fuzzy sliding controller is better than that of the conventional T-S fuzzy controller.

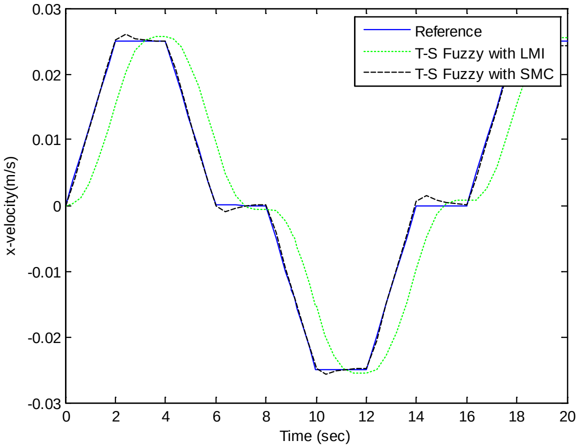


Figure 9. The variation of x-velocity

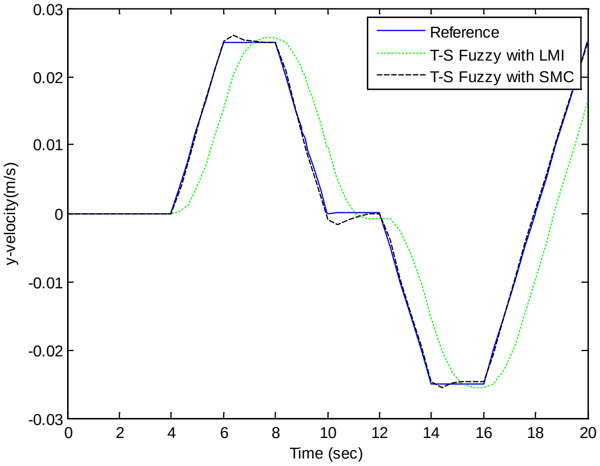


Figure 10. The variation of y-velocity

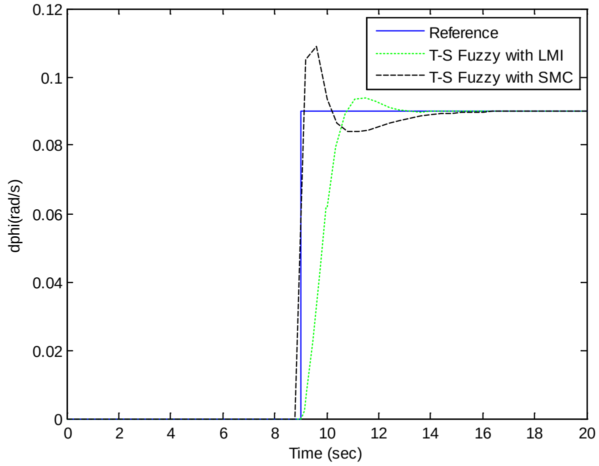


Figure 11. The variation of $\dot{\phi}$

B. Path tracking

To illustrate the effectiveness of the T-S fuzzy sliding control design, let the path tracking command be the reference signals shown in Fig. 9 and Fig. 10, respectively. The conventional T-S fuzzy control is applied to the mobile robot for comparison. Fig. 9~11 shows the responses of x-velocity, y-velocity, and angle velocity, respectively. Fig. 12 shows the path tracking response of the mobile robot. Obviously, the performance of the T-S fuzzy sliding controller is superior to that of the conventional T-S fuzzy controller.

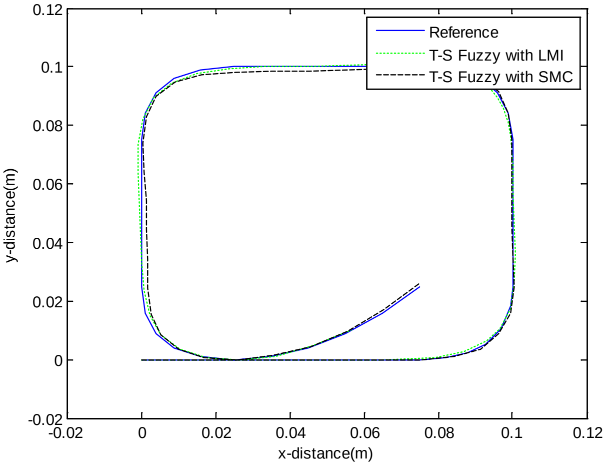


Figure 12. The response of path tracking

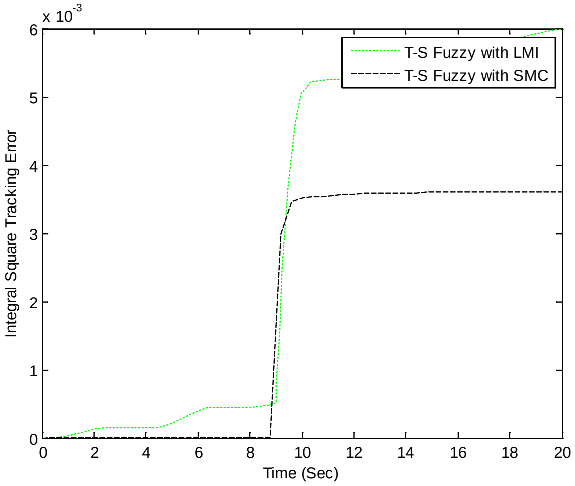


Figure 13. The integral square tracking error

Fig. 13 shows the integral square tracking error of the mobile robot. Obviously, the tracking error of the T-S fuzzy sliding controller is less than that of the conventional T-S fuzzy controller. The tracking performance is measured by

$$E_I = \int_{t=0}^{10} e^T(t)\, e(t)\, dt \qquad (31)$$

VI. Conclusion

The path controller design of a mobile robot based on T-S fuzzy model and sliding mode control has been proposed. The T-S fuzzy system was used to model the mobile robot. The PDC controller was designed by stability condition derived from Lyapunov theory. In order to promote the system performance, the sliding mode control was integrated into the T-S fuzzy system. Computer simulation demonstrates the T-S fuzzy sliding controller is better than the conventional T-S fuzzy controller for the mobile robot.

References

[1] F. G. Pin and S. M. Killough, "A new family of omnidirectional and holonomic wheeled platforms for mobile robots," IEEE Trans. on Robotics and Automation, Vol. 10, No. 4, pp. 480-489, 1994.

[2] H. Asama, M. Sato, L. Bogoni, H. Kaetsu, A. Matsumoto and I. Endo, "Development of an omni-directional mobile robot with 3 DOF decoupling drive mechanism," in Proc. of the 1995 IEEE International Conference on obotics and Automation, Aichi, Japan, pp. 1925-1930, 1995.

[3] K. Watanabe, Y. Shiraishi, S. G. Tzafestas, J. Tang and T. Fukuda, "Feedback control of an omnidirectional autonomous platform for mobile service robots," Journal of Intelligent and Robotic Systems, Vol. 22, No. 3, pp.315-330, 1998.

[4] K. Watanabe, "Control of an omnidirectional mobile robot," in Proc. of the 1998 IEEE Second International Conference on Knowledge-Based Intelligent Electronic Systems, Adelaide, Australia, pp. 51-60, 1998.

[5] C. S. Tseng, B. S. Chen and H. J. Uang, "Fuzzy tracking control design for nonlinear dynamic systems via T-S fuzzy model," IEEE Trans. on Fuzzy Systems, Vol. 9, No.3, pp. 381-392, 2001.

[6] K. Tanaka and H. O. Wang, Fuzzy Control System Design and Analysis – A Linear Matrix Inequality Approach, Wiley, NY, 2001.

[7] K. Tanaka and M. Sugeno, "Stability Analysis of fuzzy Systems Using Lyapunov's Direct Method", Proc. NAFIPS'90, pp.133-136, 1990.

[8] Y. W. Liang, S. D Xu, D. C. Liaw and C. C. Chen, "A Study of T-S Model-Based SMC Scheme with Application to Robot Control," IEEE Transactions on Industrial Electronics, Vol. 55, No. 11, pp. 3964-3971, 2008.

[9] H. K. Lam, "Stabilization of nonlinear systems using sampled-data output-feedback fuzzy controller based on polynomial-fuzzy-model-based control approach," *IEEE Trans. Syst., Man, Cybern.*, vol. 42, no. 1 pp. 258 - 267, Feb. 2012.

[10] K. Tanaka, H. Ohtake, T. Seo, M. Tanaka, and H. O. Wang, "Polynomial fuzzy observer designs a sum-of-squares approach," *IEEE Trans. Syst., Man, Cybern.*, vol. 42, no. 5, pp. 1330 - 1342, Oct. 2012.

[11] H. K. Lam, C. Liu, L. Wu and X. Zhao, "Polynomial fuzzy-model-based control systems: stability analysis via approximated membership functions considering sector nonlinearity of control input," *IEEE Trans. Fuzzy Systems*, vol. 23, no. 6, pp. 2202-2214, 2015.

[12] K. Tanaka, M. Tanaka, Y. J. Chen, and H. O. Wang, “A new Sum-of-Squares design framework for robust control of polynomial fuzzy systems with uncertainties,” *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 11, pp. 1, April. 2015.

[13] G. R. Yu, Y. C. Huang and C. Y. Cheng, “Robust H∞ controller design for polynomial fuzzy control systems by sum-of-squares approach,” *IET Control Theory Appl.*, vol. 10, no.11, pp. 1684 - 1695, Sep. 2016.

Gwo-Ruey Yu received the Ph.D. degree in Electrical Engineering from the University of Southern California, Los Angeles, in 1997. He is currently a Professor of Electrical Engineering Department and the Director of Elegant Power Application Research Center, National Chung Cheng University, Taiwan. Dr. Yu is respectively the recipients of the Best Paper Award of IEEE 2017 International Automatic Control Conference, the Advisor Award of Robotic Society of Taiwan in 2018, the Best Paper Award in Application of IEEE 2016 International Conference on Fuzzy Theory and It's Applications, the Outstanding Paper Award of IEEE 2016 International Automatic Control Conference, the Best Paper Award of 2018 International Conference on Advanced Robotics and Intelligent Systems. His research interests include intelligent robots, automatic control based on intelligence, and renewable energy systems.

Wen-Yen Chen received the B.S. degree in electrical engineering from National Chung Cheng University, Taiwan, in 2015, the M.S. degree in electrical engineering from National Chung Cheng University, Taiwan, in 2017. His research interests include intelligent robots, fuzzy systems, and intelligent control.

Terrain detection by a Self-organizing CNN

Maxwell Hwang and Kao-Shing Hwang

***Abstract*—One of the drawbacks of supervised learning for convolution neural networks (CNNs) is the lack of efficient manners to collect training features. Therefore, we proposed a self-organizing synthetic mechanism to facilitate the training ability of recognition. The proposed architecture consists of two functional modules: a self-organizing synthesis layer (SOS) embedding in the convolution neural network. The former automatically classifies and refines incoming training or testing patterns into several clusters, which are regarded as the template features in recognition stage; that is, the clustered features are present to the input layers of the latter module for training. The proposed system is applied to the terrain detection for a mobile robot, where an LED pointer projects a straight line on the terrain by an angle of 45 degrees. The image of the line is captured by a CCD camera mounted at the front of the robot. It demonstrated that the recognition rate is above 95% after 20 patterns have been synthesized and trained and above 99% after 30 patterns have been trained in the experiments.**

Index Terms*—impedance control, compliance control, compliance teaching, human-robot cooperation

I. Introduction

CONVOLUTION neural networks are a hierarchical multi-layer neural network, capable of deformation-invariant visual pattern recognition, even if the input pattern is deformed in shape, shifted in position or imposed by noises [1-2]. Corresponding to the category of the input pattern, only one cell is activated in the higher stage of the network. Other cells respond to other categories. It can acquire the ability to recognize patterns by learning and can be trained by either supervised or unsupervised learning. In practice, there is a major problem in the training process of a convolution neural network. it needs hard labor to construct a tremendous amount of good training set. To solve these problems, a two-fold neural network called Flexible Self-organizing Synthetic Convolution Neural Network (FSS-CNN) is proposed in this paper. The neural network adopts a framework of convolution neural networks but evolves with a self-organizing synthetic mechanism. The proposed architecture consists of two functional modules: self-organizing synthesizer (SoS) and a CNN. The module of SoS takes charge of pattern decomposition, feature extraction functions and clustering from the training pattern. The number of planes in each stage of the CNN hence is determined according to the number of clustered features and the clustered features are presented to the input layer of the CNN for training.

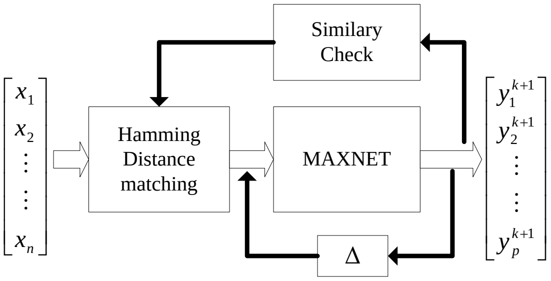


Figure 1. Diagram of the network that executed the clustering process.

II. Self-organizing Synthetic Mechanism for Kernels

There are two functions processing in the self-organizing synthetic mechanism: the decomposition/extraction and feature clustering. The feature extraction function is for the purpose of features refinery from the training patterns. The process calculates the central coordinate of a feature pattern by means of the geometry mean method and normalizes the feature pattern to meet the size of the receptive field of a "seed cell" in. The clustering is processed by a simple Hamming network with out-star connections between the input nodes and the output nodes. The connections store output features with self-organizing clustering as shown in Figure 1.

The training pattern x is a form of a binary vector in the proposed system. The Hamming network is employed to selecting a stored class that is at a minimum Hamming distance (HD) with the incoming noise or incomplete training pattern presented at the input[10]. The Hamming network consists of two layers. The first layer is Hamming Distance Calculating layer calculating the matching score between the input vector x and the stored pattern vectors in a feed-forward manner. The second layer of the Hamming network is called MAXNET, which is a recurrent recall network. The MAXNET selects the node with the maximum matching score of the first layer and suppresses the other output nodes.

If the Hamming network serves as a vector classifier with p outputs and n bits input vector x is representing training patterns. The output node connects to an input vector with a weight matrix w_m. The m's output node net_m represents the matching score between input patterns and clustered features. The equation can be expressed as

$$net_m = (w_m)^t x = n\text{-}HD(w_m, x) \qquad (1)$$

Instead of using the above equation the net_m can be expressed as

$$net_m = x^t(w_m) = \sum_{j=1}^{m} x_j \overline{w_m j} \qquad (2)$$

Maxwell Hwang is with the School of Medicine, Zhejiang University, Hangzhou, China. (mail: himax26@126.com)

Kao-Shing Hwang is with the Electrical Engineering, National Sun-Yat sen University, Kaohsiung, Taiwan. (corresponding author, mail: hwang@ccu.edu.tw)

The authors deeply acknowledge finance support from the Ministry of Science and Technology (MOST), Taiwan, ROC, under contract MOST 107-2218-E-005 -005 -

the weight vector $\overline{w_m j}$ is the normalized weight vector with small bias, which can be defined as:

$$\overline{w_{mj}} = \frac{w_{mj}}{\varepsilon + \sum_{\mathrm{j}} w_{mj}} \tag{3}$$

The ε is a bias value and set to 0.5 in this work. The purpose of Eq.(3) is to prevent the input pattern being classified into an incorrect clustered feature, which contains more useful pixels (i.e. pixel value with 1) then correct feature does.

The second layer of the Hamming network is MAXNET, whose purpose is to enhance the *m*-th node with a maximum matching score and to suppress the others. MAXNET is a recurrent network with both excitatory and inhibitory connections. The weight matrix $\mathbf{w}_{\mathrm{m}}$ of the MAXNET is the size of $p \times p$ and shown as below:

$$\mathbf{w}_{\mathrm{m}} = \begin{bmatrix} 1 & -\eta & -\eta & \dots & -\eta \\ -\eta & 1 & -\eta & \dots & -\eta \\ -\eta & -\eta & 1 & \dots & -\eta \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -\eta & -\eta & -\eta & \dots & 1 \end{bmatrix}$$

Where $0<\eta<1/p$, is called the lateral interaction coefficient. Lippmann proved that MAXNET will always converge and find the node with the maximum value when $\eta<1/p$ [11]. When initialized with the input vector, the network starts processing it by adding positive self-feedback and negative cross-feedback. When the MAXNET stabilizes, the only unsuppressed node will be the one with the largest initializing entry. This means that the only nonzero output response node is the node closest to the input vector. The recurrent update rule of MAXNET according to the equation as:

$$y^{k+1} = f(\mathrm{w_m} y^k)$$

where y^k and y^{k+1} represent output in step k and $k+1$, respectively. The activation function $f(\bullet)$ is given by

$$f(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}$$

The convolution neural network is successive to the SoS. Each stage consists of two layers: a layer of simple cells, or S-cells, followed by a layer of complex cells, or C cells. Each layer is divided into a number of planes, each of which consists of a rectangular array of cells. the S-cells respond to features at higher levels of abstraction via a variety of kernels (filters); for example, corners with intersecting lines at various angles and orientations. The C-cells integrate the responses of groups of S-cells by Max pooling operations. Because each S-cell is looking for the same feature in a different location, the C-cells' response is less sensitive to the exact location of the feature on the input layer. This behavior is what gives the Convolution neural network its ability to identify characters regardless of their exact position in the field of the retina. By the time it has reached the final layer of C-cells, the effective receptive field of each cell is the entire retina.

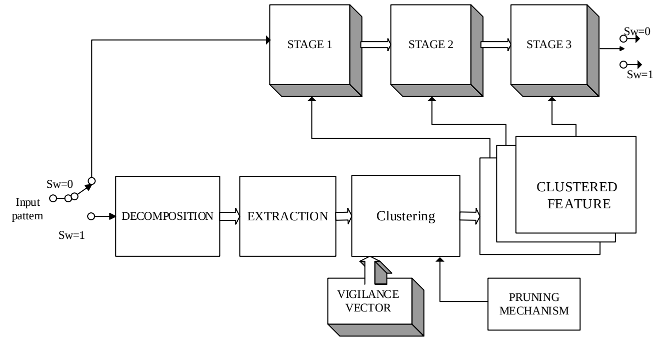


Figure 2. Framework of FSS-CNN

The original convolution neural network requires some foreknowledge of the desired features on each stage, and each plane on a stage, to respond. Under that, a set of training vectors should be developed for each layer, and the layers should be trained independently. It selects the representative cell for each plane carefully, to ensure that the input pattern is presented in the proper location with respect to the representative's receptive field [12.]

III. Flexible Self-organizing Synthetic Convolution Neural Network (FSS-CNN)

The detailed framework of FSS-CNN is shown in Figure 2, which includes two major parts: Self-organizing Synthesizer(SOSN) in the lower part and CNN in the upper part.

The self-organizing synthetic mechanism functionally consists of a decomposition, feature extraction function and clustering with a pruning mechanism. It classifies the training patterns into some clusters of features and tailors them to fit the required size to train CNN.

The decomposition and extraction function analyzes the training pattern and refines the features. The features obtained from the decomposition process are all fixed in size and some of them are not well-located on the template window. The meaning of refining a feature is to put the character distribution of the feature to the center of the pattern and to tailor the features to meet the request of training the successive network. The feature extraction function generates the central coordinates of a feature pattern by means of the geometry mean method as section 2.1 in this paper.

The purpose of the clustering process is to classify the refined features under the control parameter of vigilance vector. The vigilance vector, with 0 < V ≦ 1, measures the degree of discrimination between different classes of refined features. If V is small, the similarity condition is easier to meet, resulting in a coarse categorization and more refined patterns are merged into a category. On the other hand, if V is chosen to be close to 1, many finely divided categories are formed and fewer refined patterns are merged. The network decrease in size through a pruning process.

As an input vector lies outside sensitivity regions of all, a new neuron is added. In order to avoid incremental growth infinitely, the network decrease in size through a pruning process. The probability of an operational neuron deleted increases in direct proportion to the overlap between its sensitivity region and regions of its neighbors. The overlap between the sensitivity regions of several neurons is estimated by computing the frequency of activation of overlapping neurons with the same input vector. Therefore, the deletion probability of a neuron is increasing linearly.

The whole process is illustrated as follow:

```
Step 1: set the vigilance vector V=0.1, 0.4, 0.9.
Step 2: initialize parameters
Step 3: present a pattern vector x, where $\mathbf{x}\in\{0,1\}^n$
Step 4: according to the size of seed cells, extract the feature of
        x. and using the geometry mean method to locate central
        coordinates of each extracted feature.
Step 5: calculate the match score M($p,w_j$) between the refined
        feature and weight vector $w_j$
Step 6: if M($p,w_j$) < V for all j then
            activates a new neuron $j*$ by initializing its
            reference vector to $w_{j*}=p$, and its deactivation
            parameter to $Pr_{j^*}=1$
        else
            update the weight vector w
Step 7: if several neurons are activated in step 6, deactivate one
        of neurons if $rnd(\ ) > Pr_j$ ,where $Pr_j$ is deactivation
        parameter, and $rnd(\ )$ is a uniformly distributed random
        number in the range (0,1). The probability of
        deactivation of activated neuron is increasing as follows:
        $Pr_j(t+1) = Pr_j(t) - \eta(Pr_j(t) - Pr_{min})$, where $\eta$ and $Pr_{min}$ are
        pruning constants.
Step 8: goto step 3
```

Another part of FSS-CNN is the CNN framework. The CNN in this paper is trained by means of supervised learning. That is to say, the synaptic connections of the network are reinforced by means of a learning way with a "teacher". The "teacher" in the proposed system is the self-organizing synthetic mechanism. The number of S-cell and C-cell planes of the CNN fully depends on the number of clustered features.

The whole process of FSS-CNN in pattern recognition executed can be divided into two serial phases: a training phase and recall phase. When a new pattern is presented to the input layer of the network, in recall phase, could not be recognized, the system will get into the training phase. During training phase, the flexible self-organizing synthetic mechanism automatically extracts the shape of refined feature, tailors the size of refined feature, and also calculates the central coordinates of a refined feature. Meanwhile, it also determines the number of S-cell and C-cell planes in each stage of CNN. The feature clustering process classifies the refined features into several clusters of features. This mechanism also presents the clustered features to the CNN and chooses a "seed cell" in a plane for training those seed cell. Training of CNN is performed step by step from the lower to the higher stage. The response of the cells on the highest stage of the network shows the final results of pattern-recognition in the recall phase. After being well trained, the network can perform a rather robust pattern recognition without being affected by any deformation, such as changes in size, or shifts in position.

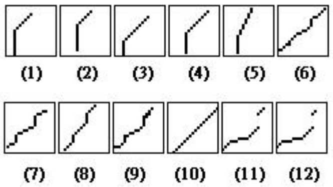
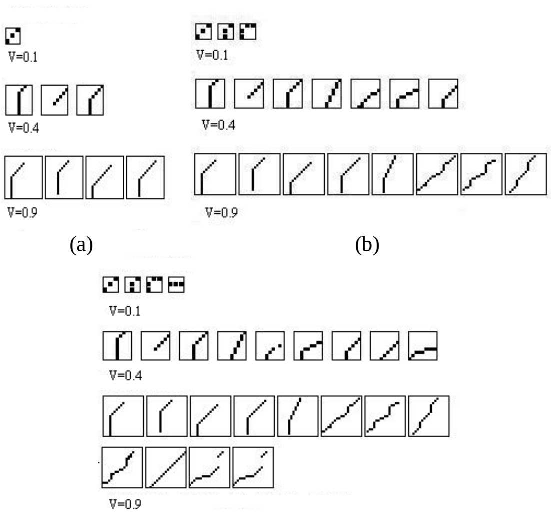


Figure 3. Training patterns in simulation 1



(a) (b)

(c)

Figure 4. Response of self-organizing synthetic mechanism
(a) 4 patterns have been presented; (b) 8 patterns have been presented;
(c) 12 patterns have been presented..

IV. Experiments

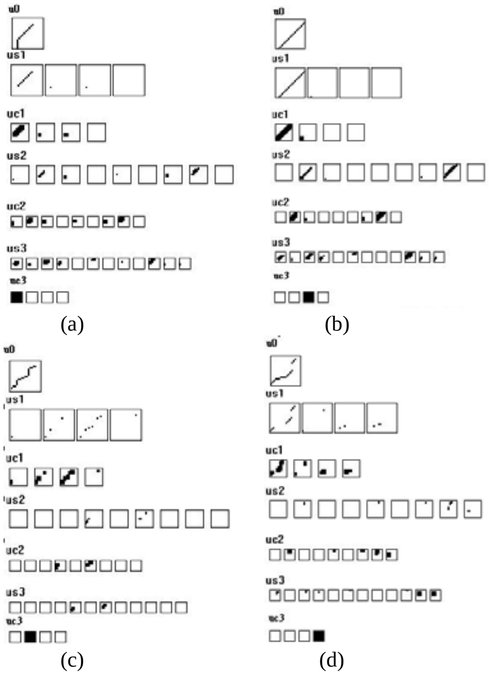
In order to test the performance of FSS-CNN, two simulations are presented. In the first simulation, we present twelve BMP training patterns; each contains 19*19 pixels'. Each of training patterns, representing the image map that a laser pointer projects on the ground by the degree of 45, is used to train CNN. In the second simulation we examine the ability of recognition to the digital numerals from 0 to 9.

The network of flexible self-organizing synthetic mechanism is presented with twelve 19*19 pixels' training patterns a shown in Figure 3. in those figures, each of them represents a laser pointer projects on the ground, pattern(1)~pattern(5) represent the "corner" of a wall; pattern(6)~pattern(9) represent the "stairs"; pattern(10) represents "clear space", and the others represent the "obstacle".

A. Simulation 1

1). Response of self-organizing synthetic mechanism

In simulation 1, after pattern(1)~pattern(4) are been presented to the flexible self-organizing synthetic mechanism, the network yields the clustered features and be shown in Figure 4(a). After pattern(5)~pattern(8) and pattern(9)~pattern(12) were presented, the responses of flexible self-organizing synthetic mechanisms are shown as Figure 4(b) and Figure 4(c), respectively.



(a) (b)

(c) (d)

Figure 5. Response of CNN (a) corner (b) space (c) stair (d) obstacle.

2). Response of CNN

The CNN, which has been trained with refined patterns, is tested by a set of patterns. In this simulation, four test pattern including "clear space", "stairs", "obstacle", and "corner" pattern are presented to the input layer of CNN. The responses of cells in CNN are shown in Figure 5 respectively.

B. Simulation 2

In this simulation, we present hand-written numerals through 0 to 9 at the input layer of FSS-CNN step by step and randomly. When a numeral pattern is presented to the input layer of the network, in recall phase, could not be recognized, the system will get into the training phase and the self-organizing Synthesizer generated the refined-clustered features according with the vigilance vector and pruning mechanism in the clustering entity.

It is difficult to state quantitatively to what degree the system can deal with deformation in patterns because we do not have an appropriate mathematical measure to correctly express the psychological feeling of the deformation. For this reason, we only could evaluate the performance of this network roughly. The recognition rate is above 95% after 20 patterns have been trained, and above 99% after 30 patterns have been trained in this simulation. Such that the numbers of clustered features almost do not increase after 30 patterns presented.

V. Discussion

The performance of the original convolution neural network is highly dependent upon the selectivity of training features and this is hard laboring work for the designed to create the features. The Proposed CNN is derived from the generic convolution neural network. The shape of training feature and the number of S-cell planes fully depend on the features, which refined and clustered by the Self-organization Synthesizer. The SOS plays an important role for CNN to recognize patterns. The SOS itself does not need to prepare the entire set of training patterns, momentarily may present pattern for training into the input layer to do for training. Therefore, FSS-CNN is a network that is capable of online learning and has a highly flexible characteristic. It is very suitable to be used in changeable terrain detection.

References

[1] K, Fukushima, "Cognitron: A Self-organizing Multilayered Neural Network." *Biol. Cybernetics*, Vol. 20, 137-143 ,1975

[2] Y. LeCun, Y. Bengio and G. Hinton; "Deep learning," Nature, vol. 521, pp. 436–444, May. 2015.

[3] Y. LeCun, et al., "Progressive inter-scale and intra-scale non-blind image deconvolution, " *ACM Transactions on Graphics*, Vol. 27, No 3, August 2008.

[4] S. Cho, et al., " Handling outliers in non-blind image deconvolution," *Proceedings of 2011 International Conference on Computer Vision*, Nov. 2011.

[5] D. Zoran, Y. Weiss, "From learning models of natural image patches to whole image restoration," *Proceedings of 2011 International Conference on Computer Vision*, Nov. 2011.

[6] G. Carpenter and S. Grossberg "A Massively Parallel Architecture for Self-organizing Neural Pattern Recognition Machine" *Computer Vision, Graphics and Image Processing*, Vol.37 ,1986 , 54-115

[7] J. P. Albright "An Implementation and Evaluation of The ART1 Neural Network for Pattern Recognition" *IEEE International Conference on Neural Networks*, 1994

[8] Carpenter, G.A. & Grossberg, S. " Adaptive Resonance Theory" T*he Handbook of Brain Theory and Neural Networks*, Second Edition ,pp. 87-90, 2003.

[9] I.-C. Jou, S.-S. Yu, & S. C. Tsay,"A New Feature Extraction Method By Neural Networks," *IEEE International Symposium on Circuits and Systems*, 1990.

[10] J, M. Zurada, Introduction to Artificial Neural System, *Info Access Distribution Pte* Ltd.

[11] R. P. Lippmman, "An Introduction to Computing with Neural Nets," *IEEE Acoustics, Speech, and Signal Processing*, April 1987.

[12] D. R. Lovell and A. C. Tsoi,"An Evaluation of The Convolution neural network," *IEEE Transactions on Neural Networks*, Vol. 8, no. 5, September 1997.

Maxwell Hwang is a Ph.D. candidate of Zhejiang University, School of Medicine. He received the M.D. degree in Zhejiang University School of Medicine, Hangzhou, Zhejiang, China, in 2015. His current research interest includes the applications of artificial intelligence in colonoscopy, Colorectal Cancers, Laparoscopic Surgery, and medical image analysis.

Kao-Shing Hwang (M'93–SM'09) is a professor of Electrical Engineering Department at National Sun Yat-sen University, and an adjunct professor of the department of Healthcare Administration and Medical Informatic, Kaohsiung Medical University, Taiwan. He received the M.M.E. and Ph.D. degrees in Electrical and Computer Engineering from Northwestern University, Evanston, IL, U.S.A., in 1989 and 1993, respectively. He had been with National Chung Cheng University in Taiwan from 1993-2011. He was the deputy director of Computer Center (1998-1999), the chairman of the Electrical Engineering Department (2003~2006), and the director of the Opti-mechatronics Institute of the university (2010~2011) in National Chung Cheng University. He has been a member of IEEE since 1993 and a Fellow of the Institution of Engineering and Technology (FIET). His research interest includes methodologies and analysis for various intelligent robot systems, machine learning, embedded system design, and ASIC design for robotic applications.

2D Indoor Occupancy Grid Mapping for Mobile Robotic Platforms Using ROS-based ORB-SLAM2

Emiliano Alban, *Student Member, IEEE*, Chin-Sheng Chen. *Member, IEEE*, and Chia-Jen Lin

***Abstract*—This manuscript covers a new frontier to ascertain whether Visual Simultaneous Localization and Mapping (vSLAM) has the ability, by itself, to perform further motion tasks as path planning and autonomous navigation by using a single RGB-D camera. The proposed solution is the creation in real-time of a standard Two-Dimensional Occupancy Grid Map (2D OGM) module for an indoor environment leveraging complete compatibility with the Robot Operating System (ROS). The robot pose is estimated by the state-of-the-art feature-based algorithm so-called Oriented FAST and Brief SLAM 2 (ORB-SLAM2). Subsequently, the map is built by the combination of the depth information and the pose of the mobile robotic platform (MRP). The experimental process is divided into two stages: virtual and real environments. The purpose of the first part is to collect data and gain expertise configuring the system, whereas the second stage delineates the possibility to implement the suggested approach in a real application. Therefore, the resulting map is compared with a ground truth map constructed by a ROS LIDAR-based method to validate the potential and system's applicability. Furthermore, this study aims to transmit essential knowledge to those roboticists who are beginning to delve a viable vSLAM-based solution.**

Index Terms*—Visual Simultaneous Localization and Mapping (vSLAM), Robot Operative System (ROS), ORB-SLAM2, 2D Occupancy Grid Map, RGB-D Camera, Mobile Robotic Platform applications, Depth image, Gazebo Simulation, RViz Monitoring.

I. Introduction

MOBILE robotic platforms (MRP's) possess significant and rising applications in academic, commercial, and industrial fields such as novel algorithms development, shipments, warehousing, and logistics robots, to name a few. Nowadays, Robot Operating System, better known as ROS, has become the de-facto standard open-source framework for programming robotics to build effective, flexible, and expandable solutions including a well-known problem called Simultaneous Localization and Mapping (SLAM) [1], [2].

SLAM encompasses a couple of simultaneous requirements: the estimation of the state of a robot (*localization*) and its surroundings modeling (*mapping*) by utilizing data acquired from onboard sensors. This research focuses on the mapping capability that comprises three concerns. First of all, the primary necessity of a map creation of an unknown environment. Secondly, the built map supports further tasks as path planning, autonomous navigation, and even it might proffer visual information for a human operator. Thus, the error computed in the estimation of the robot state is considerably decreased by the use of a consistent map and the place recognition adroitness know as loop-closure detection, which solves the last concern.

The proposed approach shares the perspective of Cadena *et al.* [3] that defines three eras in the historical advancement of SLAM. The *classical age* (1986-2004) covered the essential formulations for SLAM characterized by extended Kalman Filters (EKF), Rao-Blackwellized particle filters, and maximum likelihood estimation. The posterior era is denominated as *algorithmic-analysis age* (2004-2015) which examined observability, convergence, and consistency properties of SLAM. Furthermore, the efficient and main open-source SLAM libraries were elaborated in this period. At present, the *robust-perception age* is taking place in SLAM studies and is delineated by four essential criteria: robust performance, high-level understanding, resource awareness, and task-driven perception. All of them are still unsolved challenges. In the interest of contributing with a primer solution, this research implements a feature-based method known as Oriented FAST and Brief SLAM 2 (ORB-SLAM2) which in many instances such as [4] [5], [6], [7] and [8] has been established as the state-of-the-art feature-based algorithm founded on its accuracy, robustness, automatic relocation and loop closure capability.

Moreover, as it is argued in [9], the ORB feature descriptors evidence the best performance in several aspects as quantity feature detection, computational efficiency, feature-matching efficiency per feature-point, and total matching speed rate. The back-end output of the method mentioned above is a sparse-points 3D map. Hence, the limitations and drawbacks of utilizing this outcome to link up into other ROS packages, the solution proposed is the creation in real-time of a standard Two-Dimensional Occupancy Grid Map (2D OGM) module for an indoor environment as a typical office surrounding. The depth image provided from an RGB-D camera is converted into a standard laser-scan message, which combined with the MRP's pose estimated by the vSLAM algorithm and rectified by a Transformation Broadcaster, allows the map creation. Subsequently, the resulting map is compared with the map built by a ROS LIDAR-based method denominated as gmapping[1], an improved approach presented in [10] that applies Rao-Blackwellized particle filters to decrease the possibility of particle depletion and loop closure detection. Besides, it possesses high accuracy distribution

E. Alban and C.-J. Lin are with the Graduate Institute of Mechanical and Electrical Engineering, National Taipei University of Technology, Taipei, 10608, Taiwan (e-mail: emiliano_alban@outlook.com, t107669011@ntut.edu.tw).

C.-S. Chen is with the Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, 10608 Taiwan (e-mail: saint@mail.ntut.edu.tw).

This study was financially sponsored by TECO Electric & Machinery Co., Ltd. under project number 208N01.

[1] gmapping package: http://wiki.ros.org/gmapping

concerning the observation likelihood of the last received sensor data, the odometry, and a scan-matching process. The map created by gmapping has elected as ground truth following the results described in [11] and [12] pointing it out as the best performance evaluated method.

In the beginning, this manuscript summarizes the works related to the available map builders based on ORB-SLAM2. The following section contains the key concepts and mathematical background the research comprises to develop the proposed system. Thence, the simulated and real environments are described in detail to move forward into the map comparison and finally establishing the conclusions and further research.

II. Related work

The vSLAM research community is continuously growing and formulating novel applications for MRP, in particular, applying open-source SLAM libraries as ORB-SLAM2 and benchmarks to analyze their improvements, but for applications in real environments, there are only a few.

Mo, Chen, Wang, and Wang [13] proposes an autonomous exploring mobile robot annexing a new thread in the original system [4]. This additional function can build a 3D local octree dense map using point cloud information acquired from an RGB-D camera. Then, the octree-map is projected into a 2D occupancy grid mapping in order to proceed with the exploration execution. Following a similar methodology, a navigation system for obstacle avoidance in humanoid robots is performed by He, Leng, and Zha [14]. Both types of research have no evidence or more in-depth information that shows the 2D OGM quality obtained in their systems to compare with the solution proposed in this study.

Quite the opposite occurs in the 2D OGM module proposed recently by Xu, Feng, Kamat, and Menassa [15], which combines the robot pose estimated by ORB-SLAM2 with the laser-scan converted from the point cloud information obtained from the camera. Additionally, they have achieved path planning and continuous navigation with full ROS compatibility on homogeneous corridors and laboratory rooms at school buildings. Due to the recentness of their publication, it does not exist an open-source package available to develop any further experimental tests and analyzes. Dissimilar to the methodology aforementioned, the present manuscript asserts the map creation based on depth information instead of point cloud data and the experimental tests are performed in a fully-equipped office environment. It is worth mentioning that both studies were developed almost at the same time.

III. Background

A. Robot Operating System

ROS [2] is a nontraditional operative system (OS) for management and scheduling process; instead of that, it administers and maintains a well-structured communications framework over the OS host in a heterogeneous computing network. Furthermore, it supports large-scale software integration and code reuse. Philosophically, the purpose of ROS is based on five essential pillars: peer-to-peer, tools-based, multi-lingual, slim, and free-open source platform [16].

The fundamentals of the Robot Operative System are constructed in four main concepts: nodes, messages, topics, and services [1], [16].

- *Nodes* are instances on ROS that execute computations allowing the OS being modular. They communicate to each other through messages can be classified as publisher and subscriber nodes, e.g., when a publisher node transmits data, the information provided is available for any subscriber node which must be able to establish communications with the publisher. Note that a node can be subscriber and publisher at the same time.
- *Messages* are strictly typed packets that can be composed by other messages. There are many types of messages, from standards primitives such as integer, floating point, boolean, and so forth to arrays of primitive types, constants or a twist used to describe three-dimensional and rotational velocities.
- *Topics* are the strings to define a particular sort of data that allows proper communication among publisher and subscriber nodes.
- *Services* are synchronized and remote procedure calls to transmit data between nodes designed to work recurrently to perform computations in a limited amount of time.

The openness and versatility of ROS architecture have allowed the creation of several tools that may be categorized into eight segments, as is shown in the next section:

1) *High-end drivers support:* on ROS, interfacing any sensor or actuator requires no effort. Robotics device drivers and packages are available and maintained by their official developers.
2) *Debugging nodes:* for developing robotics research and applications, the system involved is frequently a vast scheme. To simplify its management, ROS has designed a modular structure which is able to insert and remove nodes at runtime in the most natural way possible increasing notably the efficiency and productivity when a robotic system becomes more and more complex.
3) *Logging and playback:* any topic and its data associated can be logged into the disk in order to be replayed in next sessions to perform controlled comparisons and modifications of algorithms in development, making plainer, and further experimental procedures.
4) *Packaged subsystems:* this feature is oriented to the ability of ROS for running an entire process composed of many nodes with one single command typing, a tool called *roslaunch.*
5) *Cooperative Development:* at the time of writing, there are more than 2500 official ROS packages properly documented, maintained, and assisted by an active community of users and developers [2], [17].
6) *Visualization and Monitoring:* ROS Visualization (RViz) might be one of the more powerful and useful tools that this ecosystem has developed dynamically to view several sorts of data - depth images, laser scans, geometric primitives robot poses, trajectories, and

[2] ROS link: http://wiki.ros.org/

suchlike. All of them, configurable instantaneously from a panel in the RViz plugin.

7) *Composition of functionality:* elaborated systems as mobile-robot navigation might require multiple cluster-nodes initializations with total certainty of no data collisions involved. ROS grants it through the design of software named *stuck*, which is a set of nodes that working altogether perform an important task.

8) *Transformations:* tf (short for transform) is a ROS package to manage spatial relationships and its transforms. The tf system builds a dynamic tree based on all the frames referenced in the coordinates system.

B. ORB-SLAM2

Mur-Artal and Tardós in [4] describe an integral and open-source SLAM system that supports monocular, stereo and RGB-D cameras exposing real-time, loop closing and relocation accomplishments, also includes drift reduction, metric scale, and ROS package support[3]. As is illustrated in Fig. 1, the global framework of the ORB-SLAM2 system is principally formed by three threads working in parallel: tracking, local mapping, and loop closing, the latter, enables the creation of a secondary thread to accomplish full Bundle Adjustment (BA) and map updating afterward a loop closure detection. The next section comprehends a brief description of the central part of the threads developed in [18].

The algorithm estimates the camera pose in the actual frame using ORB feature points that match with the previous frame performing an optimization pose using motion-only BA. Subsequently, the matched feature points in the local map sparse-points are searched by reprojection allowing the camera pose optimization with all matches. In the end, this thread decides whether the insertion of a new keyframe should take place or not. Motion-only BA enhances the camera orientation $R \in SO(3)$ and position $t \in \mathbb{R}^3$, decreasing the reprojection error between matched 3D points $X^i \in \mathbb{R}^3$ in world coordinates and key points $x^i_{(.)}$, either monocular $x^i_m \in \mathbb{R}^2$ or stereo $x^i_s \in \mathbb{R}^3$, with $i \in X$ denominated the set of all matches such that:

$$\{R,t\} = \underset{R,t}{\arg\min} \sum_{i\in X} \rho\left(\left\| x^i_{(\cdot)} - \pi_{(\cdot)}\left(RX^i + t\right)\right\|^2_{\Sigma}\right) \qquad (1)$$

In (1), ρ corresponds to the robust Huber cost function and Σ is the covariance matrix associated with the key point's scale.

The projection functions $\pi_{(.)}$, represented as π_m and π_s for monocular and stereo respectively are defined in (2) and (3), where (f_x, f_y) are the focal lengths, (c_x, c_y) correspond to the principal point in the image, and b represents the camera baseline.

$$\pi_m = \left(\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}\right) = \begin{bmatrix} f_x \dfrac{X}{Z} + c_x \\ f_y \dfrac{Y}{Z} + c_y \end{bmatrix} \qquad (2)$$

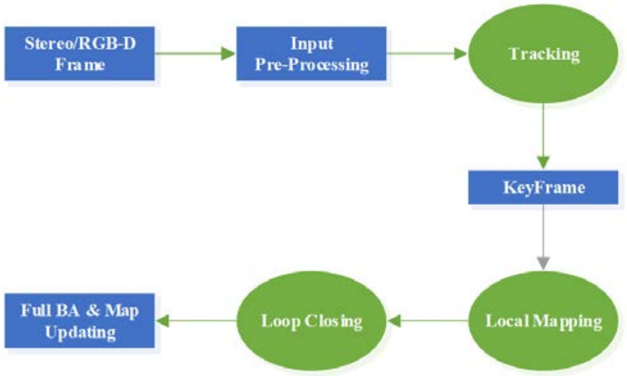


Figure 1. The global framework of the ORB-SLAM2 system.

$$\pi_s = \left(\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}\right) = \begin{bmatrix} f_x \dfrac{X}{Z} + c_x \\ f_y \dfrac{Y}{Z} + c_y \\ f_x \dfrac{X-b}{Z} + c_x \end{bmatrix} \qquad (3)$$

C. Occupancy Grid Maps

In mobile robotics, an occupancy grid map is a classic and frequently used method to represent the robot's surroundings. This representation, proposed by Moravec and Elfes [19], [20] ordinarily makes use of two or three juxtaposed dimensions to transform a real space into an array of cells, where each cell contains information of the occupancy probability and the spatial uncertainty of the environment. The smaller the spatial uncertainty, the lesser the number of cells occupied by any object [21].

Thrun [22] points out that two algorithm actions distinguish the standard occupancy mapping techniques:

- The computation of numerous binary estimation problems, obtained by a decomposition of a highest-dimensional mapping problem, is resolved independently one from another. Moreover, the formation of conflicts created by the decomposition is inevitable. This drawback is treated on the grid cell level by the utilization of Bayesian methods. Paradoxically, even the noise-free data suffer the same issue.
- The map is generated by using inverse models, meaning that the map's information comes from the robot's sensors.

The creation of a map m and its accuracy relies on the set of sensor measurements, $z^t=\{z_1,\ldots,z_t\}$, such as sonar or lidar data and the path of the robot, $x^t=\{x_1,\ldots,x_t\}$, structured as the sequence of all posses in which the acquisition data was performed. The algorithm benchmark to solve any occupancy grid mapping problem is based on the computation of the posterior over maps given the measurements z^t.

$$p\left(m \mid z^t, x^t\right) \qquad (4)$$

[3] Original implementation: https://github.com/raulmur/ORB_SLAM2

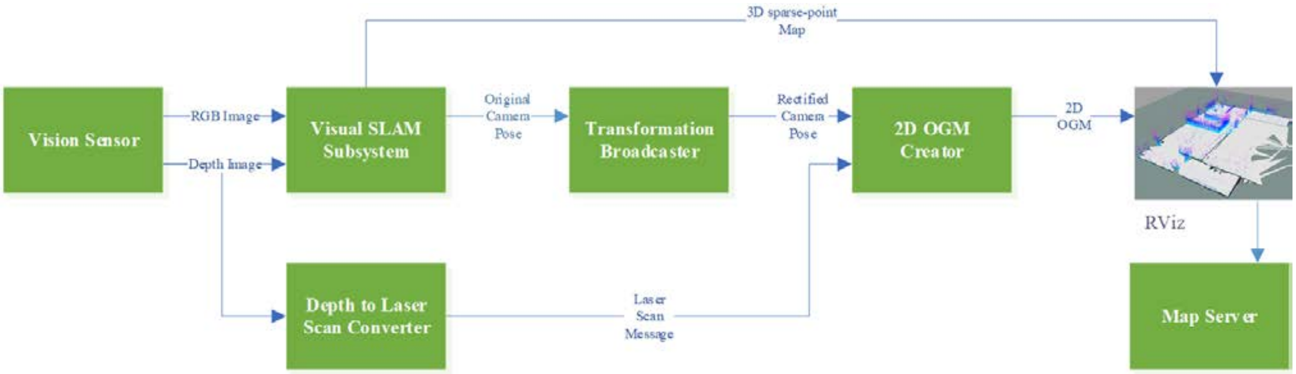


Figure 2. Overview of the 2D Indoor Occupancy Grid Mapping System.

Occupancy grid maps are considered fine-grained grids determined throughout the continuous space which contains finitely but plenty of grid cells, denoted as m_i:

$$m = \{m_i\} \qquad (5)$$

In (5), a binary occupancy value is assigned to each m_i, implying whether a grid cell is unoccupied or not. It is assigned a "0" for unoccupied and "1" for occupied. Hence, as it has mentioned before, the occupancy grid approach is analyzed as a cluster of independent problems:

$$p\left(m_i \mid z^t, x^t\right) \qquad (6)$$

The posterior over maps can be expressed by the product of the probability computed in each independent grid cell, namely that of marginals:

$$p\left(m \mid z^t, x^t\right) = \prod_i p\left(m_i \mid z^t, x^t\right) \qquad (7)$$

TABLE I summarizes the algorithm that Thrun, Burgard, and Fox propose in [23] to solve the occupancy grid mapping problem applying a binary Bayes filter and the log-odds representation of occupancy:

$$l_{t,i} = \log \frac{p\left(m_i \mid z^t, x^t\right)}{1 - p\left(m_i \mid z^t, x^t\right)} \qquad (8)$$

The probability representation might have numerical instabilities for values close by one or zero; this drawback can be avoided through the application of the log-odds, and the probabilities are computed from the log-odds ratio:

$$p\left(m_i \mid z^t, x^t\right) = 1 - \frac{1}{1 + \exp\left\{l_{t,i}\right\}} \qquad (9)$$

The function $ism(m_i, x_t, z_t)$ represents the inverse sensor model $p(m_i \mid z_t, x_t)$ utilizing the log-odds form:

$$ism\left(m_i, x_t, z_t\right) = \log \frac{p\left(m_i \mid z_t, x_t\right)}{1 - p\left(m_i \mid z_t, x_t\right)} \qquad (10)$$

The constant l_0 is known as prior of occupancy:

$$l_0 = \log \frac{p\left(m_i = 1\right)}{p\left(m_i = 0\right)} = \log \frac{p\left(m_i\right)}{1 - p\left(m_i\right)} \qquad (11)$$

TABLE I.
STANDARD OCCUPANCY GRID MAPPING ALGORITHM USING THE BINARY BAYES FILTER

```
Std_Occ_Grid_Algrthm({$l_{t-1,i}$}, $x_t$, $z_t$):
    for all grid cells $m_i$ do
        if $m_i$ in the perceptual range of $z_t$ then
            $l_{t,i} = l_{t-1,i} + ism(m_i, x_t, z_t) - l_0$
        else
            $l_{t,i} = l_{t-1,i}$
        endif
    endfor
    return {$l_{t,i}$}
```

IV. System Overview

As shown in Figure 2, the 2D Indoor Occupancy Grid Mapping System is composed of five modules: Visual SLAM (ROS-based ORB-SLAM2), Transformation Broadcaster, Depth Image to Laser Scan Converter, 2D OGM Creator, and the Map-Server. Although the proposed architecture has performed experiments in simulated and real environments by using only RGB-D cameras, this approach can also use stereo cameras as an input system.

Real-time 2D OGM construction and monitoring are visualized on RViz[4] software. Besides, the 3D sparse-point map generated by the visual odometry module is added and synchronized into the RViz environment. The occupancy grid map might be saved into the storage disk using the Map-Server Module.

The TF Broadcaster builds a proper TF tree to obtain a rectified camera pose to construct the map, correctly. All of the modules as mentioned earlier are full-compatible ROS packages enabling the reuse and interconnections with other developed systems on ROS as Navigation or Motion Planners Stacks to achieve autonomous navigation tasks.

A. Vision Sensor

To generate a 2D OGM, the system requires two different sorts of images as input: RGB and depth. Both of them acquired from the Vision Sensor Module (VSM) emphasizing that both frames must be aligned. In other words, their topics must have the same timestamp in ROS to establish a proper correspondence between their coordinate systems. This

[4] Rviz package: http://wiki.ros.org/rviz

condition should be satisfied in order to achieve an accurate estimation of the MRP's pose and the map construction.

This study employs two Intel RealSense Cameras that generate aligned images, the models R200 and D435 for simulation and real environments, respectively.

B. Visual SLAM

This module computes the camera pose and generates a 3D sparse-point map using a ROS-based ORB-SLAM2 package. The original implementation developed in [4] does not possess any compatibility with the standard navigation ROS messages making unachievable any further application or research development such as motion planning or autonomous navigation. To address this inconvenient, the system utilizes an alternative package[5], which has removed the Pangolin dependency and added the input and output manipulation data via ROS topics leveraging full compatibility and configuration.

The Visual SLAM Module (VSLAMM) possesses two types of parameters related to the vision sensor and the algorithm itself. The first-mentioned are those related to the camera so-called intrinsic parameters, which are based on the pinhole camera model and may be obtained after performing a camera calibration. TABLE II summarizes a description of the intrinsic camera parameters.

The second type of parameters configure and calibrate the VSLAMM performance and has two sub-classifications: static and dynamic. Several parameters belong to the static type; all of them have to be cautiously configured before the system starts to work. They cannot be set up at runtime being this the reason to be named as static parameters. TABLE III shows the description of the critical static parameters.

On the contrary, there are only three dynamic parameters that can be configured at runtime. TABLE IV contains their description.

C. Transformation Broadcaster

The coordinate system so-called frame of the camera pose computed by the VSLAMM does not correspond with the standard 3D right-handed frame on ROS that has the next axis directions: X forward, Y left, and Z up. Usually, the world coordinate system is known as *map*, which is the referenced coordinate system to subsequently obtain all the frames poses in a robot. The relationship between two frames is a frame pose, which is composed by a translation vector, and a rotation matrix represented via bullet quaternions.

TABLE II.
INTRINSIC CAMERA PARAMETERS

No.	Parameter	Description	Units
1	f_x, f_y	Focal lengths expressed in pixel (px) units.	(px)
2	c_x, c_y	Coordinates of the principal point that is often located at the image center.	(px)
3	k_1, k_2	First two higher-order radial distortion coefficients.	(units)
4	p_1, p_2	Tangential distortion coefficients.	(units)
5	bl	Camera baseline: the distance between right and left imager.	(m)

TABLE III.
ORB-SLAM2 STATIC PARAMETERS DESCRIPTION

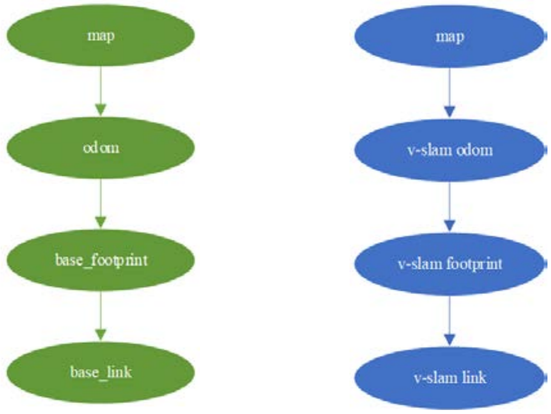
No.	Parameter	Description	Units
1	fps	Frame rate: number of unique and consecutive frames that the system is set to handle.	(frame/ sec)
2	bf	Multiplication's result of the x-axis focal length (fx) by the camera base line (bl).	(px x m)
3	ThDepth	Depth threshold to define how far or close the system looks for ORB features in the image. It is multiplied by the camera baseline in order to obtain a distance threshold.	(units)
4	nFeatures	Maximum number of ORB features allowed per image.	(units)
5	width	Width camera resolution.	(px)
6	height	Height camera resolution.	(px)

TABLE IV.
ORB-SLAM2 DYNAMIC PARAMETERS DESCRIPTION

No.	Parameter	Type	Description
1	mode	boolean	It defines the operation mode of the system: 0 for SLAM and 1 for only localization.
2	reset	boolean	It allows a re-initialization system to erase the original map.
3	min_keyframes	integer	It sets the minimum number of keyframes on the map to keep the system works with no auto-reset.

Transformation Broadcaster Module (TFBM) has two objectives to achieve in the system. The first purpose is to modify the camera pose's coordination system in order to guarantee total correspondence with the ROS standard frame. The second objective is to preserve the relationship between the coordinate frames in the MRP through the time utilizing a transformation (TF) tree, as illustrated in Figure 3.

TFBM makes use of the *tf package*[6] to build a tf tree, as shown in Figure 3 (b) which follows the standard hierarchical structure developed for mobile robots for LIDAR-based SLAM, as illustrated in Figure 3 (a).



(a) Lidar-based TF tree (b) Visual Odometry based TF tree

Figure 3. Typical TF tree on ROS

[5] ORB-SLAM2 ROS: http://wiki.ros.org/orb_slam2_ros

[6] tf package: http://wiki.ros.org/tf

TABLE V.
PARAMETERS OF THE DEPTH IMAGE TO LASER SCAN CONVERTER

No.	Parameter	Description	Units
1	scan_height	Number of pixel-rows to create the virtual laser scan.	(px)
2	scan_time	The sampling time for scanning. Usually, the value is given by the expression: 1/fps.	(sec)
3	range_min	Minimum threshold distance	(m)
4	range_max	Maximum threshold distance	(m)
5	output_frame_id	Topic name to identify the created virtual laser scan.	NA

D. *Depth Image to Laser Scan Converter*

ROS environment utilizes a specific type of message identified as *sensor_msg/LaserScan* for acquiring 2D laser data arrays from measurements performed from laser scanners. This message and the robot's pose become the required inputs to create a 2D OGM and update the map periodically.

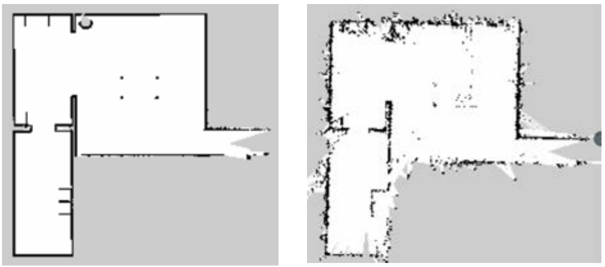
Similarly, in the Depth Image to Laser Scan Converter Module (DI2LSCM), a laser scan message is obtained using the depth image originated in the RGB-D camera node. The conversion process is handled by an open-source package named *depthimage_to_laserscan*[7] that employs a depth image to create a virtual laser scan by extracting a horizontal segment of the image given a pre-defined height and other parameters, all of them summarized in TABLE V.

E. *2D Occupancy Grid Map Creator*

The 2D OGM Creator Module (2DOGMCM) subscribes to the topics published by the TFBM and the DI2LSCM receiving the MRP's pose and the virtual laser scan respectively to execute the creation and recurring update of the map. The map's quality relies directly on two factors: accuracy of the robot's pose and proper calibration of the 2DOGMCM parameters. The last to be mentioned is related to the occupancy grid map algorithm detailed in the literature review section and the map updating parameters listed in TABLE VI.

TABLE VI.
PARAMETERS OF THE 2D OCCUPANCY GRID MAP CREATOR.

No.	Parameter	Description	Units
1	map_update_rate	Map-updating rate. The lower the rate, the more computational load is required.	(sec)
2	linear_update	Set how long the MRP needs to move to process a new scan.	(m)
3	ang_update	Set how much the MRP needs to rotate to process a new scan.	(rad)
4	temp_update	Define the scan process mode. The value 1 sets the mode based on the angular and linear update. The value -1 enables the scan process in the case that the time taken between the last two scans is greater than the map-updating rate.	(unit)
5	occ_threshold	Set value of a cell's log-odds at the first exploration time.	(%)



(a) LIDAR-based (b) vSLAM-based

Fig. 4. An example of a map saved by Map-Server.

To achieve a creation's map with high quality, the parameter rates such as linear, angular, and map updating must be configured as smaller as they can to maintain the computational load in a reasonable level meaning that an experimental calibration may be implemented to accomplish a high-performance system.

F. *Map-Server Module*

An essential function of every mapping system is based on the capability to storage the maps created from each visited place. The aim of the Map-Server Module (MSM) is developing tasks such as map analysis, metric accuracy comparisons, path planning, and autonomous navigation.

The proposed system applies a ROS package known as *map_server*[8] belonging to a more extensive system called ROS Navigation Stack[9] which owns complete support for 2D planning and navigation tasks. The map server might be used as a service or a command-line tool to manipulate map data. An example of a saved map looks is shown in Fig. 4.

The illustration portrays an occupancy grid map in which the spatial uncertainty of each cell corresponds to a standard pixel color configurations and ternary values: white means unoccupied (0), black is for occupied (100), and gray denotes unvisited or unknown state (-1).

TABLE VII.
PUBLISHED TOPICS TO VISUALIZE ON RVIZ.

No.	Topic	Module Publisher	Description
1	color_image	VSM	It shows the MRP's environment in RGB format.
2	depth_image	VSM	It displays the MRP's environment using depth format.
3	debug_image	VSLAMM	The ORB features detected are shown in this image combined with a grayscale version of the image environment.
4	vslam_footprint	VSLAMM	It represents the pose estimated with no standardization of the ROS coordinate system.
5	pointcloud_map	VSLAMM	The 3D sparse-point metric map that contains all the ORB features detected by the vSLAM Module.
6	vslam_link	TFBM	A standardized ROS robot's pose.
7	depthlaser_scan	DI2LSCM	It shows the virtual laser scan created from the depth image and the MRP's pose.
8	2doccgr_map	2DOGMCM	Through this topic, the creation of the map is displayed in real-time.

[7] Laser scan converter: http://wiki.ros.org/depthimage_to_laserscan

[8] Map server package: http://wiki.ros.org/map_server

[9] Navigation stack: http://wiki.ros.org/navigation

G. *Visualization and Supervision*

This approach becomes a versatile system regarding visualization and supervision functionalities drawing on a popular ROS tool as RViz. The monitoring window has been set in such a manner that the user might manipulate and interact with the data obtained from the topics published by the system modules thus depth image, point clouds, poses, maps among other topics. TABLE VII collects the critical topics that the user needs to configure in real-time to have an appropriate visualization of the whole system.

Figure 5 exhibits the monitoring interface that primarily is composed by the combination of the 3D sparse-point map and the 2D occupancy grid map enabling to contemplate the robustness and accuracy of the proposed system.

V. Implementation and Experiments

First, several experiments were held in a virtual scenario to validate the functionality, robustness, and accuracy of the vSLAM method applied to estimate the MRP pose. It is performed to analyze the viability of applying the proposed system in a real environment. This section depicts the system setup, environmental descriptions, intrinsic camera parameters, and tuning map parameters.

A. *System Setup*

1) *Virtual environment:* For this stage, we implemented a virtual ROS-based mobile robot called TurtleBot3 [10] (TB3) that presents the small size and embedded system properties. Equally important, this device has expandable and customizable characteristics as well as broad support in Gazebo Simulator [11] that makes it a versatile and affordable option for new researchers. The main features of the TB3 are summarized in TABLE VIII, that includes the technical characteristics of the personal computer in which the simulations were carried out. It should be noticed that the information showed in the next table corresponds to the model implemented in the virtual environment.

TABLE VIII.
HARDWARE AND SOFTWARE SPECIFICATIONS OF THE MASTER AND HOST STATIONS IN THE VIRTUAL ENVIRONMENT.

Parameter	***Description***	
	Virtual TB3	***PC-Master***
	Hardware	
Processor	Intel Joule 570x	Intel Core i5-7300HQ
GPU	N/A	GeForce GTX 1050
RAM	2 Gb	16 Gb
	Software	
Operative System	Ubuntu 16.4	Ubuntu 16.4
ROS Version	Kinetic	Kinetic
	Onboard Sensors	
2D LIDAR	360 Laser Distance Sensor LDS-01	N/A
Camera	Intel RealSense R200	N/A

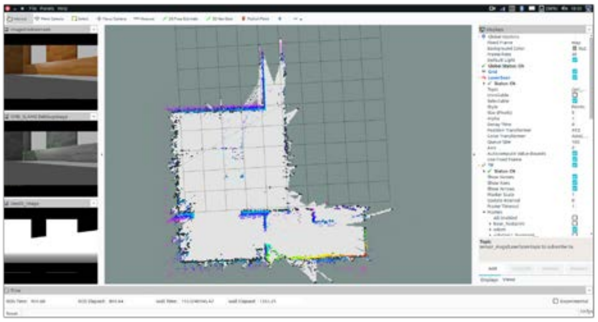


Figure 5. Visualization and Supervision of the 2D OGM System.

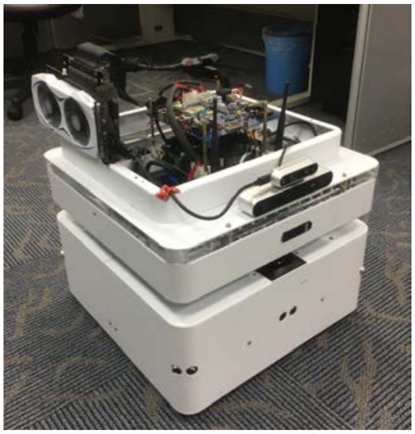


Figure 6. Mobile robotic platform.

2) *Real environment:* The implemented ROS network has quite a standard master-host configuration. The MRP runs as ROS master computer. Figure 6 illustrates the implemented prototype of a differential-drive ground vehicle equipped with an onboard industrial computer, many sensors, and it is designed to move through indoor scenarios.

The remote PC is a standard laptop utilized to teleoperate the MRP, supervise the process, and visualize the map construction. TABLE IX condenses the leading hardware and software specifications of the master and host stations.

TABLE IX.
HARDWARE AND SOFTWARE SPECIFICATIONS OF THE MASTER AND HOST STATIONS IN THE REAL ENVIRONMENT.

Parameter	***Description***	
	MRP-Master	***Remote PC-Host***
	Hardware	
Processor	Intel Core i7-6700TE	Intel Core i5-7300HQ
GPU	GeForce GTX 1050 Ti	GeForce GTX 1050
RAM	16 Gb	16 Gb
	Software	
Operative System	Ubuntu 16.4	Ubuntu 16.4
ROS Version	Kinetic	Kinetic
	Onboard Sensors	
2D LIDAR	Hokuyo UAM-05LP	N/A
Camera	Intel RealSense D435	N/A

[10] TB: http://emanual.robotis.com/docs/en/platform/turtlebot3/overview/
[11] Gazebo Simulator: http://wiki.ros.org/gazebo_ros_pkgs

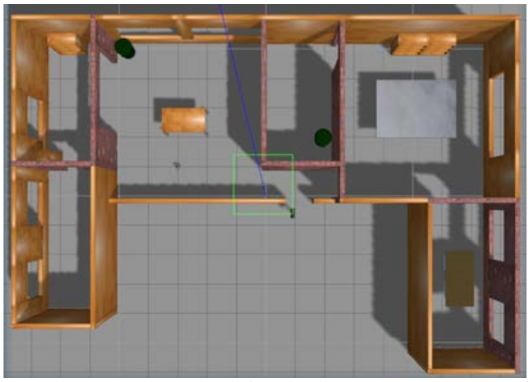


Figure 7. Top view of the virtual environment.

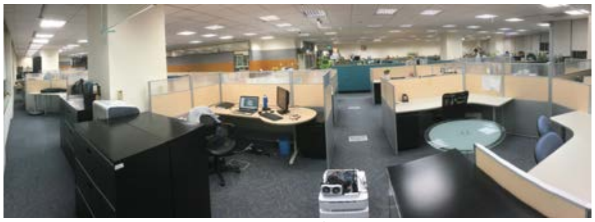


Figure 8. Panoramic view of the real environment.

B. Environment Description

1) *Virtual environment:* It was adapted as a one-floor house scenario with six rooms containing some furniture such as tables, cabinets, and a couple of trashcans. The area covered by this environment is up to 15m×10m. A top view of this building is shown in Figure 7.

2) *Real environment:* The entire set of experiments and calibrations have been conducted in an office environment with plenty of furniture. Figure 8 illustrates how heterogeneous and complex is the MRP surrounding, making the map building a real challenge. The area explored has the dimensions 7.5 m x 10 m and contains eleven workstations with two longitudinal and three transversal corridors. The light level is constant, and the walking people flow rate is moderate.

C. Configuration of the Intrinsic Camera Parameters

The performance of the system relies on the proper assignation of the intrinsic camera parameters due to the geometry and metrics are directly related to them. Indeed, an incorrect configuration may cause the robot's rotation or translation to become inaccurate, generating an erroneous map. The next table presents the values configured to develop the experiments in the virtual and real environments.

D. Map Creator Tuning

Once the intrinsic camera parameters configuration is adequately performed, then the map creator requires a refinement procedure known as tuning process based on several trial tests, and researcher's knowledge and expertise. TABLE XI abstracts the setup parameters of the system. They might be used as a guide to replicate the results obtained in these experiments.

TABLE X.
CONFIGURED INTRINSIC CAMERA PARAMETERS

		Virtual Environment	***Real Environment***	
No.	**Parameter**	**Value**		**Units**
1	f_x	~1206.9	~616.2	(px)
2	f_y	~1206.9	~616.4	(px)
3	c_x	~960.5	~317.9	(px)
4	c_y	~540.5	~246.3	(px)
5	p_1	0	0	(units)
6	p_2	0	0	(units)
7	k_1	0	0	(units)
8	k_2	0	0	(units)
9	bl	0.036	0.05	(m)

TABLE XI.
TUNING SYSTEM PARAMETERS

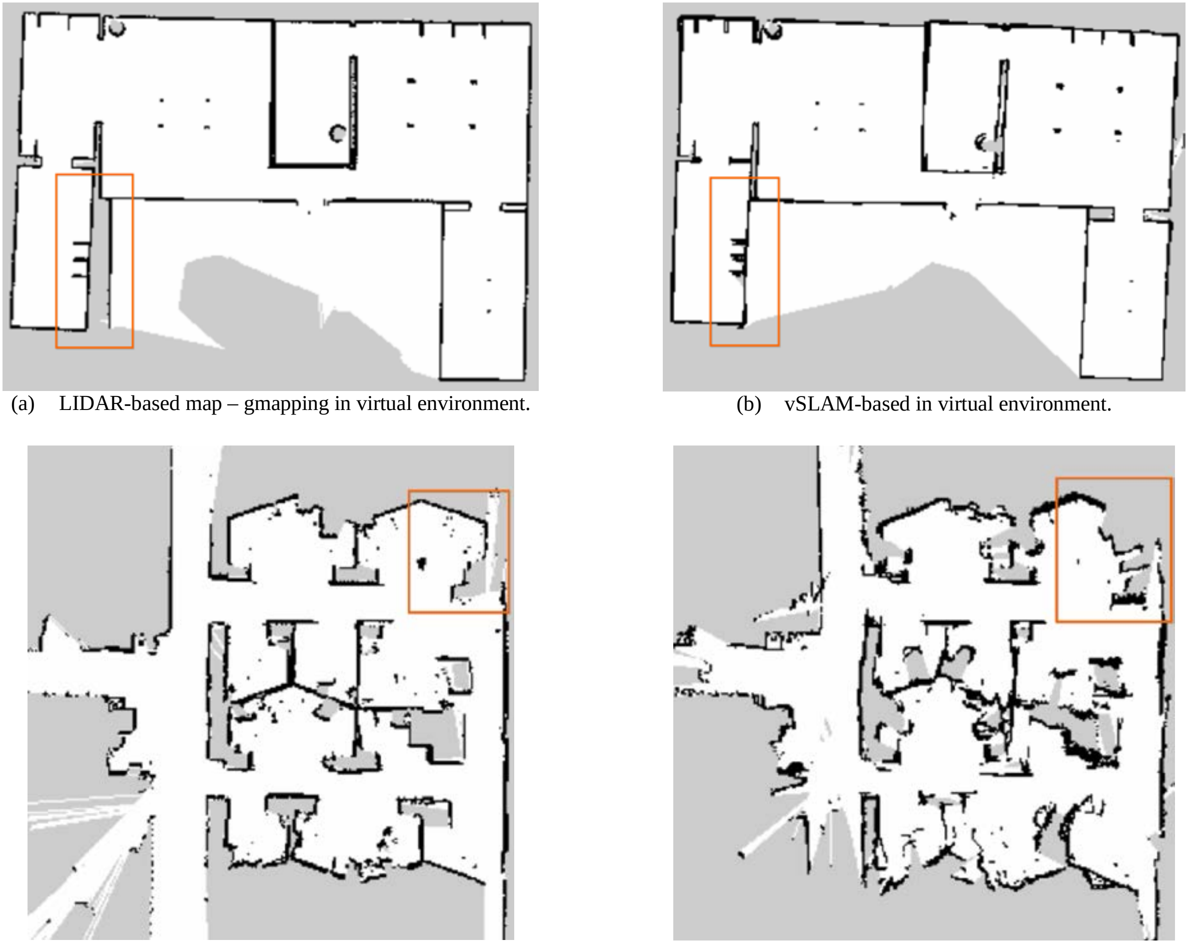
		Virtual Environment	***Real Environment***	
No.	**Parameter**	**Value**		**Units**
Visual SLAM Module				
1	fps	30	30	(px)
2	bf	~43.3	~30.8	(px)
3	ThDepth	10	40	(px)
4	nFeatures	1200	1000	(px)
5	width	1920	640	(units)
6	height	1080	480	(units)
Depth Image to Laser Scan Module				
1	scan_height	10	50	(px)
2	scan_time	~0.33	~0.33	(sec)
3	range_min	0.4	0.11	(m)
4	range_max	15.0	3.0	(m)
2D Occupancy Grid Map Creator Module				
1	map_update_rate	2	0.25	(sec)
2	linear_update	0.005	0.04	(m)
3	ang_update	0.001	0.005	(rad)
4	temp_update	1	1	(unit)
5	occ_threshold	0.11	0.25	(%)

VI. RESULTS AND DISCUSSION

The first output of the system is a superposition of two maps: the 3D sparse-point cloud obtained from the VSLAMM and the map plotted by the 2D Occupancy Grid Map Creator.

As is shown in Figure 10, both maps hold a close metric relationship to each other. It is expected given that the grid map creation (black line), is key-based on the MRP pose estimated by the vSLAM Subsystem that also generates the 3D sparse-point map (colored points).

The maps saved by the Map-Server Module are arranged in Fig. 9. The above images (a) and (b) corresponds to the maps in the virtual environment experiments obtained by LIDAR-based and vSLAM-based methods correspondingly. The orange rectangles enclose a mapped area where two walls are located side by side, as seen in Figure 7.



(a) LIDAR-based map – gmapping in virtual environment.

(b) vSLAM-based in virtual environment.

(c) LIDAR-based map – gmapping in real environment.

(d) vSLAM-based in real environment

Figure 9. 2D occupancy grids saved by the Map Server Module.

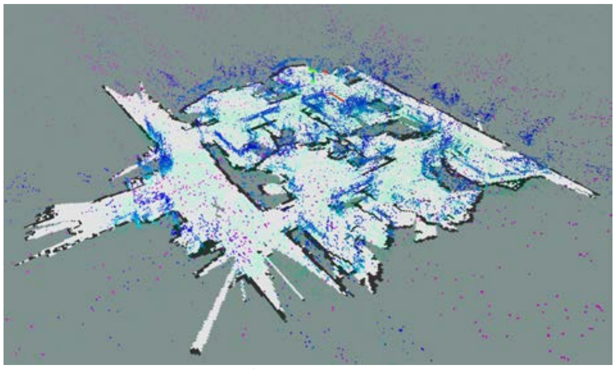


Figure 10. Superposition of the 3D sparse-point and 2D occupancy grid maps.

The approach proposed is the only one able to create a map, which describes correctly that environmental configuration. The lower images (c) and (d) exhibit the resulting maps built in the real environment by the LIDAR-based and vSLAM-based techniques, respectively. The orange square spotlights the section in the real environment illustrated in Figure 11 and implicitly highlights an asset of the map creator to plot more in detail the MRP's surroundings providing more accurate and realistic information of the explored environment. Indeed, this advantage could contribute to improving avoidance collision functions in tasks such as path planning and autonomous navigation becoming safer the MRP applications development.

However, the vSLAM method presents significant drawbacks once the MRP purely moves at high rotation velocity. It directly incurs the occasional loss of the robot pose. In order to prevent it, TABLE XII contains the recommended velocities for the robot's teleoperation.

VII. Conclusions

A 2D indoor occupancy grid system has been proposed in this manuscript using a ROS-based ORB-SLAM2 for indoor environments. The modular system advantages the accuracy performed on the estimation of the mobile robotic platform computed by the visual-SLAM Subsystem that combined with the depth information acquired from the RGB-D camera achieve a robust mapping system. In fact, this study has demonstrated the high-viability to develop further indoor applications in virtual and real environments as a result of the more detailed information contained in the maps created by the system in this research.

Although the reached approach evidenced applicability (via qualitative methods) to accomplish tasks like obstacle avoidance or motion planning, the authors will conduct a quantitative method to analyze the accuracy of the metrics

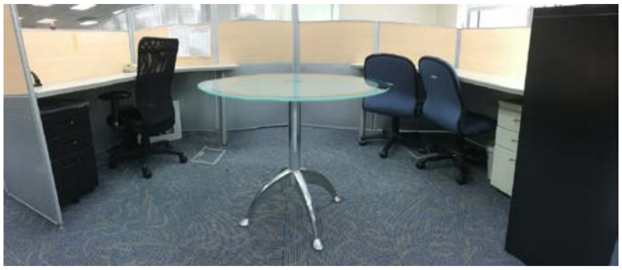


Figure 11. Heterogeneous area in the real environment.

TABLE XII.
RECOMMENDED VELOCITIES FOR MRP OPERATION.

No.	Motion	Range	Value	Units
1	Pure Translation Linear Velocity	Min	0.2	(m/s)
2		Max	0.8	(m/s)
3	Pure Rotation Angular Velocity	Min	0.1	(rad/s)
4		Max	0.24	(rad/s)

providing more solid sustentation to this approach. Therefore, an object labeling function will be performed to add semantic information derived from the robot's surrounding.

Finally, this paper aims to be used by the future members of the SLAM community as an initial guide in their studies encouraging them to enter in such a challenging path as the development of visual-SLAM applications in real environments.

REFERENCES

[1] W. S. Newman, A systematic approach to learning robot programming with ROS, Boca Raton, Florida, U.S.: CRC Press, 2018.

[2] M. Quigley, B. Gerkey and W. D. Smart, Programming robots with ROS, Sebastopol, California, U.S.: O'Really Media Inc., 2015.

[3] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Towards the robust-perception age," *IEEE Transactions on Robotics,* vol. 32, no. 6, pp. 1309-1332, 2016.

[4] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics,* vol. 33, no. 5, pp. 1255-1262, 2017.

[5] B. M. da Silva, R. S. Xavier, T. P. do Nascimento and L. M. Gonçalves, "Experimental evaluation of ROS compatible SLAM algorithms for RGB-D sensors," in *Latin American Robotics Symposium (LARS)*, Curitiba, Brazil, 2017.

[6] Z. Chen, W. Sheng, G. Yang, Z. Su and L. Baojuan, "Comparison and analysis of feature method and direct method in visual SLAM technology for social robots," in *World Congress on Intelligent Control and Automation*, Changsha, China, 2018.

[7] M. Filipenko and I. Afanasyev, "Comparison of various SLAM systems for mobile robot in an indoor environment," in *IEEE International Conference on Intelligent Systems (IS)*, Madeira, Portugal, 2018.

[8] R. Giubilato, S. Chiodini, M. Pertile and S. Debei, "An experimental comparison of ROS-compatible stereo visual SLAM methods for planetary rovers," in *IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, Rome, Italy, 2018.

[9] S. A. Khan Tareen and Z. Saleem, "A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK," in *International Conference on Computing, Mathematics, and Engineering Technologies (iCoMET)*, Sukkur, Pakistan, 2018.

[10] G. Grisetti, C. Stachniss, and W. Burgard, "Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resampling," in *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005.

[11] F. Anton, F. Artyom, K. Kirill, C. Baian and M. Diana, "2D SLAM quality evaluation methods," in *Conference of Open Innovations Association FRUC*, Helsinki, Finland, 2017.

[12] R. Yagfarov, M. Ivanou, and I. Afanasyev, "Map comparison of lidar-based 2D SLAM algorithms using precise ground truth," in *International Conference on Control*, *Automation, Robotics and Vision (ICARCV)*, Singapore, Singapore, 2018.

[13] H. Mo, X. Chen, K. Wang and H. Wang, "Autonomous localization and mapping for mobile robot based on ORB-SLAM," *Proceedings of Chinese Intelligent Systems Conference,* vol. 1, pp. 749-760, 2018.

[14] Z. He, X. Leng, and F. Zha, "ORB-SLAM based humanoid robot location and navigation system," *Vibroengineering PROCEDIA,* vol. 17, pp. 118-123, 2018.

[15] L. Xu, C. Feng, V. R. Kamat, and C. C. Menassa, "An occupancy grid mapping enhanced visual SLAM for real-time locating applications in indoor GPS-denied environments," *Automation in Construction,* vol. 104, pp. 230-245, 2019.

[16] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, Kobe, Japan, 2009.

[17] L. Joseph and J. Cacace, Mastering ROS for robotics programming, 2nd ed., Birmingham, UK: Packt Publishing, 2018.

[18] R. Mur-Artal, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics,* vol. 31, no. 5, pp. 1147-1163, 2015.

[19] H. P. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *IEEE International Conference on Robotics and Automation*, St. Louis, MO, USA, 1985.

[20] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer,* vol. 22, no. 6, pp. 46-57, 1989.

[21] J. J. Leonard and D.-W. H. F, "Dynamic map building for autonomous mobile robot," in *EEE International Workshop on Intelligent Robots and Systems, Towards a New Frontier of Applications*, Ibaraki, Japan, 1990.

[22] S. Thrun, "Learning occupancy grids with forward models," in *International Conference on Intelligent Robots and Systems*, Maui, HI, USA, 2001.

[23] S. Thrun, W. Burgard, and D. Fox, "Occupancy grid mapping," in *Probabilistics Robotics*, London, England, The MIT Press, 2006, pp. 284-287.

Emiliano Alban Naranjo received the B.Eng. degree in Mechatronics Engineering from Universidad de las Fuerzas Armadas ESPE, Quito, Ecuador, in 2013, and the M.Sc. degree in Mechanical and Automation Engineering from National Taipei University of Technology, Taipei, Taiwan R.O.C., in 2019.

He is a first-year Ph.D. Candidate with National Taipei University of Technology, Taipei, Taiwan R.O.C., and also an Intern Researcher in TECO Group Research Institute with TECO Electric and Machinery Co. Ltd. His research addresses the problems of mapping, navigation, and AI applications for autonomous mobile robots.

Chin-Sheng Chen received the Ph.D. degrees in mechanical engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1999. He was a researcher of Sintec Technology Co. Ltd. during 1999 to 2000 and a R&D manager of TECO Electric & Machinery Co. Ltd. from 2000 to 2002. In 2002, he joined the Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan, as an Assistant Professor. Presently he is a distinguished professor of Graduate Institute of Automation Technology at National Taipei University of Technology. His research interests include intelligent motion control, robotics, mechatronics and machine vision. He received the Outstanding Research Award from the College of Mechanical and Electrical Engineering, Taipei TECH in 2013 and 2014. Prof. Chen has published over 150 journal and conference papers and book chapters on the research.

Chia-Jen Lin received the M.Sc. degree in 2004 from National Yang-Ming University, Taipei, Taiwan, where he is currently working toward the Ph.D. degree in National Taipei University of Technology, Taipei, Taiwan.

He is also an R&D manager in TECO Group Research Institute with TECO Electric and Machinery Co., Ltd. His research interest is AGV control and service robot techniques.

Information for Authors

Aim and Scope

The *i***Robotics** is an official journal of Robotics Society of Taiwan (RST) and is published quarterly. The *i***Robotics** will consider high quality papers that deal with the theory, design, and application of intelligent robotic system, intelligent artificial system, and extension theory systems ranging from hardware to software. Survey and expository submissions are also welcome. Submission of a manuscript should indicate that it has been neither published, nor copyrighted, submitted, accepted for publication elsewhere (except that the short version may have been presented at the conferences). Submitted manuscripts must be typewritten in English and as concise as possible.

Process for Submission of a Manuscript

The *i***Robotics** publishes two types of articles: regular papers and technical notes. All contributions are handled in the same procedure, where each submission is reviewed by an associate editor who makes an initial decision to send the manuscript out for peer review or to reject without external review. Articles can be rejected at this stage for a variety of reasons, such as lack of novelty, topics outside the scope of the Journal, flaws in the scientific validity, or unprofessional presentation. We are therefore not normally able to provide authors with feedback on rejected manuscripts. If the associate editor believes the article may be of interest to our readers, it is then sent out for external peer review by at least two external reviewers. According the recommendation of the associate editor, the Editor-in-Chief makes the final decision. All manuscripts should be submitted electronically in Portable Document Format (PDF) through the manuscript submission system at [http://www.rst.org.tw]. The corresponding author will be responsible for making page proof and signing off for printing on behalf of other co-authors. Upon acceptance of a paper, authors will be requested to supply their biographies (100 to 200 words) and two copies of the final version of their manuscript (in DOC format and in PDF format).

Style for Manuscript

Papers should be arranged in the following order of presentation:

1) First page must contain: a) Title of Paper (without Symbols), b) Author(s) and affiliation(s), c) Abstract (not exceeding 150 words for Papers or 75 words for Technical Note, and without equations, references, or footnotes), d) 4-6 suggested keywords, e) Complete mailing address, email address, and if available, facsimile (fax) number of each author, f) Preferred address for correspondence and return of proofs, and g) Footnotes (if desired).
2) The text: Submitted manuscripts must be typewritten double-spaced. All submitted manuscripts should be as concise as possible. Regular papers are normally limited to 26 double-spaced, typed pages, and technical notes are 12 double-spaced, typed pages. Please see the Page charge for those who want to submit long papers.
3) Acknowledgements of financial or other support (if any).
4) References: References should be numbered and appear in a separate bibliography at the end of the paper. Use numerals in square brackets to cite references, e.g., [15]. References should be complete and in the style as follows.
   [1] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller - Part I," *IEEE Trans. Syst. Man Cybern.*, vol. 20, no. 2, pp. 404-418, 1990.
   [2] C. Golaszewski and P. Ramadge, "Control of discrete event processes with forced events," in *Proc. of 26th IEEE Conf. Decision and Control*, Los Angeles, CA, pp. 247-251, Dec. 1987.
   [3] P. E. Wellstead and M. B. Zarrop, *Self-Tuning Systems*, New York: Wiley, 1991.
   [4] Project Rezero, available at http://rezero.ethz.ch/project_en.html (last visited: 2017-07).
5) Tables
6) Captions of figures (on separate sheet of paper)

Style for Illustrations

1) It is in the author's interest to submit professional quality illustrations. Drafting or art service cannot be provided by the Publisher.
2) Original drawings should be in black ink on white background. Maximum size is restricted to 17.4 by 24.7 cm. Glossy prints of illustrations are also acceptable.
3) All lettering should be large enough to permit legible reduction of the figure to column width, sometimes as small as one quarter of the original size. Typed lettering is usually not acceptable on figures.
4) Provide a separate sheet listing all figure captions, in proper style for the typesetter, e.g., "Fig. 5. The error for the proposed controller."
5) Illustrations should not be sent until requested, but authors should be ready to submit these immediately upon acceptance for publication.

Page Charges

After a manuscript has been accepted for publication, the author's company or institution will be approached with a request to pay a page charge to cover part of the cost of publication. The charges include:

1) NT$ 5000 for the 10 printed pages of a full paper or for the 6 printed pages of a short paper, and the excess page charge of NT$ 1500 per extra printed page for both full and short papers.
2) For color figures or tables, an additional cost will be charged. The cost, depending on the number of color figures/tables and the final editing result, will be given upon the acceptance of this paper for publication.

Copyright

It is the policy of the RST to own the copyright of the technical contributions. It publishes on behalf of the interests of the RST, its authors, and their employers, and to facilitate the appropriate reuse of this material by others. Authors are required to sign a RST Copyright Form before publication.

Manuscripts (in PDF Format) Submission Website: http://www.rst.org.tw
Editor-in-Chief: Prof. Ching-Chih Tsai, Department of Electrical Engineering, National Chung Hsing University, Taiwan
Email: cctsai@nchu.edu.tw
Prof. Tzuu-Hseng S. Li, Department of Electrical Engineering, National Cheng Kung University, Taiwan
Email: thsli@mail.ncku.edu.tw
Managing Editor: Dr. Feng-Chun Tai, Department of Electrical Engineering, National Chung Hsing University, Taiwan
Email: fctai@nchu.edu.tw

iRobotics

VOLUME 2, NUMBER 3 SEPTEMBER, 2019

CONTENTS

REGULAR PAPERS

EtherCAT-based Impedance Control and Force Compliance Teaching of a 6-DOF Industrial Robotic Manipulator 1
Ching-Chih Tsai, Chun-Chieh Chan, Chun-An Lin, and Feng-Chun Tai

Mobile Robot Localization in Outdoor Environments by the Integration of GPS, IMU and Visual Odometry 11
Guo-Sheng Cai, Shih-Fen Kao, and Huei-Yung Lin

Autonomous Social Robot with Moods 17
Zong-Ze, Wu, Yu-Ting Hsiao, Xin-Yi Jian and Li-Chen Fu

LMI-based Tracking Control of a Omni-directionalMobile Robot Using T-S Fuzzy Sliding Mode 27
Gwo-Ruey Yu and Wen-Yen Chen

Terrain detection by a Self-organizing CNN 34
Maxwell Hwang and Kao-Shing Hwang

2D Indoor Occupancy Grid Mapping for Mobile Robotic Platforms Using ROS-based ORB-SLAM2 38
Emiliano Alban, Chin-Sheng Chen, and Chia-Jen Lin

TECHNICAL NOTE

iRobotics
Volume 2, Number 3, SEPTEMBER, 2019