Framework Design using the Robotic Augmented Reality for the Cyber-Physical System

Lam Phi Nguyen
Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City Vietnam
Vietnam National University Ho Chi Minh City (VNU-HCM), Ho Chi Minh City Vietnam

Ha Quang Thinh Ngo
Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City Vietnam
Vietnam National University Ho Chi Minh City (VNU-HCM), Ho Chi Minh City Vietnam

In the context of Industry 4.0, many advanced technologies have been investigated to enhance the high performance for the presently robotic system. In this paper, a novel concept to communicate between cyber model and physical hardware via augmented reality (AR) technique is introduced. In respect to hardware configuration, both forward kinematics and inverse kinematics are described to control the robot correctly. In addition, some components in practical hardware are briefly depicted. For the cyber framework, the control algorithm as well as simulation software are established in the virtual environment. To validate the proposed design, the cyber model and physical hardware using our framework are simulated and experimented. From these results, it can be seen clearly that our approach is effective and feasible for the practical applications.

Keywords: *Robotics, augmented reality, human-robot interaction, Unity, motion control.*

1. INTRODUCTION

Over the past few years, various robotic control architectures have been developed to interact with humans in different ways. It depends on which types of robots [1-3] could be deployed such as fully autonomous robot, or partially autonomous robot. Together with the increasing advancement of sensing, inference, and modelling, manufacturing tasks have been gained a novel level of automation instead of manual labour [4, 5]. According to these changes, it leads to reduce social cost and improve productivity.

Addressing these shortcomings requires ongoing research and development, improvements in AR technology and close collaboration should be noticed to create robust, reliable, and user-friendly solutions. The applications of AR-based system present too much information of environment at once while the critical factors in communication are not focused. Also, they lack of dynamic control from robotic platform, and a framework in order to manage whole system. Hence, our motivation is to propose a potential framework of AR-based integration for the human-robot interaction.

2. PREVIOUS WORKS

To summarize research on augmented reality, several topics are described in Table 1. In general, the robotic applications depend on the installation of the hardware platform for the augmented reality [6, 7]. In some studies, those techniques focus on the overlaying an object or anchoring a device to the body of robot. In [8], the robotic face was added to animate an expressive emotion in front of the mobile AR device. In contrast, other developments have enhanced robots with wearable devices that are integrated into the surrounding environment, Denavit-Hartenberg convention which were well-suited for serial manipulation, were deployed to control. The mobile shape-changing interface [9, 10] could be an example in this category with the overlaid animation of visual data. They directly augmented to interact with the working environment via additional information.

Alternatively, the AR technique enhances the surrounding context so that both physical elements such as wall, floor or ceiling and virtual object in 3D space provide supplementary information to drive robot. Compared to conventional approaches [11-14], the augmented robot brings more interest and more convenience to customers. For instance, the projection mapping method is often used to mark or augment the neighbouring objects to support the communication of robot [15]. To predict the intention of any motion, it allows to share the contents among humans or objects in the outdoor scenarios. In the other approach, the projector-mounted robots are usually deployed to augment the physically surrounding environment. In [16, 17], a wheeled robot with one projector which is attached in the body robot moves around the working area.

Presently, there is a trend to blend both the cyber and physical worlds into one platform. Many advanced examples, i.e. autonomous cars [18], wearable equipment [19], or haptic devices [20], mostly actuate our real world to make it more dynamic and reconstructed. Those blending works derive from different fields such as voice-based communication, gesture-based control or fusing techniques in order to produce more superior system. The existing approaches lack of dynamical computation, hardware configuration and partially virtual model. Therefore, a novel idea for the AR-based control via mutual communication between software and mechanical platform is introduced. With the common hardware for robotic application, it is possible to launch its cyber model in the 3D workspace.

Received: April 2024, Accepted: June 2024
Correspondence to: Ha Quang Thinh Ngo, Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet Street, District 10, HCMC, Vietnam, E-mail: nhqthinh@hcmut.edu.vn
doi: 10.5937/fme2403506N

3. PRELIMINARY ANALYSIS

To initialize our study, it is essential to analyse the structure of robot, the forward kinematic and inverse kinematic of mechanical framework, electric connection and peripheral device. The target platform is an industrial manipulator with five DOFs (Degree Of Freedoms) which is described as Fig. 1. In each link, separated coordinate has been attached in the local origin. According to the Denavit-Hartenberg conven–tion, some parameters are listed as Table 2.

where,

x_i, y_i, z_i : local coordinate of link ith

a_i : the distance between z_{i-1} axis and z_i axis along the x_i –axis

α_i : the required rotation of the x_{i-1}-axis about the x_i -axis to parallel the z_i –axis

d_i : the distance between x_{i-1} axis and x_i axis along the z_{i-1}-axis

θ_i : the required rotation of x_{i-1}-axis about the z_{i-1}-axis to parallel the x_i –axis

d_{tool} : the distance from the center of the 4th coordinate to the center of the end-effector

In general, the transformation matrix ${}^{i-1}_{i}T$ to trans–form coordinate frames B_i to B_{i-1} is represented as a product of 4 basic transformations using parameters of link (i) and joint i-1 as below

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

3.1 Forward kinematic

With the D-H table as above, transformation matrices are shown in this section. It is noticed that the purpose of forward kinematic is to estimate the information of end-effector from the rotation or displacement of each link. For the relation between link 0 and link 1, its transformation matrix ${}^{0}_{1}T$ is

$$ {}^{0}_{1}T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1\cos\alpha_1 & \sin\theta_1\sin\alpha_1 & a_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1\cos\alpha_1 & -\cos\theta_1\sin\alpha_1 & a_1\sin\theta_1 \\ 0 & \sin\alpha_1 & \cos\alpha_1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 50\cos\theta_1 \\ \sin\theta_1 & 0 & \cos\theta_1 & 50\sin\theta_1 \\ 0 & -1 & 0 & 358.5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2) $$

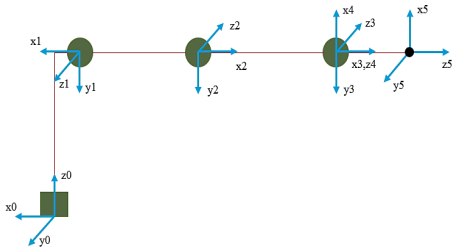


Figure 1. Theoretical diagram of our robot

This transformation matrix ${}^{1}_{2}T$ is

$$ {}^{1}_{2}T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2\cos\alpha_2 & \sin\theta_2\sin\alpha_2 & a_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2\cos\alpha_2 & -\cos\theta_2\sin\alpha_2 & a_2\sin\theta_2 \\ 0 & \sin\alpha_2 & \cos\alpha_2 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ = \begin{bmatrix} \cos\theta_2 & \sin\theta_2 & 0 & 300\cos\theta_2 \\ \sin\theta_2 & -\cos\theta_2 & 0 & 300\sin\theta_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) $$

Similarly, the corresponding transformation matrices from link 2 to link 5 are

$$ {}^{2}_{3}T = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3\cos\alpha_3 & \sin\theta_3\sin\alpha_3 & a_3\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3\cos\alpha_3 & -\cos\theta_3\sin\alpha_3 & a_3\sin\theta_3 \\ 0 & \sin\alpha_3 & \cos\alpha_3 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & 350\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & 350\sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4) $$

Table 1. Summary of the related studies in these domains

Method	Author(s)	Description	Pros	Cons
Command code	Cacace, J. et al [21]; Angleraud, A. et al [22]	Based on the known set of high-level actions, the utilization of commands was deployed to share a collaborative task	It allows one person to coordinate when and which robot reacts as soon as possible	In the environment of practical industry, noise and some falties may occur
Voice/Tone	Korayem, M. H. et al [23]	A robotic system using the voice detection method based on the hidden Markov model was designed	It is possible to extend the dictionary, acoustic, and language models of the system.	This platform is not proper for human-robot interaction
Visual signal	Dianatfar, M. et al [24]	The training program via visual and graphical components was invented to perceive assembly sequence	It increases human trust, do not require experience and actively handle the fear of human	The test case is very simple in comparing to the practical scenario

Gesture	Guzzi, J. et al [25]	A platform for the interactive demonstration was developed to simulate the automation system	This tool is very flexible and extendable	Using pointing gestures sensed by wrist-worn inertial measurement unit might not relfect the physical factors fully
Data fusion	Yongda, D. et al [26]	Human-robot interaction was enhanced by the maximum entropy classifier for speech command and Leap Motion model for hand gesture	These results from fusing data are better than unique method using either speech or gesture	More complicated commands for various tasks are needed
Virtual signal	Malik, A. A. et al [27]	Some advancements in visualization technologies might be helpful for the virtual reality	Less time to design, optimizing the simulation, and better visualization the workstation are advantageous	It does not permit to interact with natural motion of hand as well as perform the real feelings in the assembly task

Table 2. List of the robotic parameters in D-H table

Link	a_i (mm)	α_i (degree)	d_i (mm)	θ_i (degree)
1	50	-90	358.5	θ_1
2	300	180	0	θ_2
3	350	0	0	θ_3
4	0	-90	0	θ_4
5	0	0	d_{tool}	θ_5

$$
{}^{3}_{4}T = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4\cos\alpha_4 & \sin\theta_4\sin\alpha_4 & a_4\cos\theta_4 \\ \sin\theta_4 & \cos\theta_4\cos\alpha_4 & -\cos\theta_4\sin\alpha_4 & a_4\sin\theta_4 \\ 0 & \sin\alpha_4 & \cos\alpha_4 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} \cos\theta_4 & 0 & -\sin\theta_4 & 0 \\ \sin\theta_4 & 0 & \cos\theta_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)
$$

$$
{}^{4}_{5}T = \begin{bmatrix} \cos\theta_5 & -\sin\theta_5\cos\alpha_5 & \sin\theta_5\sin\alpha_5 & a_5\cos\theta_5 \\ \sin\theta_5 & \cos\theta_5\cos\alpha_5 & -\cos\theta_5\sin\alpha_5 & a_5\sin\theta_5 \\ 0 & \sin\alpha_5 & \cos\alpha_5 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 1 & d_{tool} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)
$$

Hence,

$$
{}^{0}_{5}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T
$$

$$
= \begin{bmatrix} R11 & R12 & R13 & Px \\ R21 & R22 & R23 & Py \\ R31 & R32 & R33 & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)
$$

where

$$
{}^{0}_{5}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T
$$

$$
R11 = \cos(\theta_4+\theta_3-\theta_2)\cos(\theta_1)\cos(\theta_5) - \sin(\theta_1)\sin(\theta_5)
$$

$$
R12 = -\cos(\theta_4+\theta_3-\theta_2)\cos(\theta_1)\cos(\theta_5) - \sin(\theta_1)\cos(\theta_5)
$$

$$
R13 = -\sin(\theta_4+\theta_3-\theta_2)\cos(\theta_1)
$$

$$
Px = 350\cos(\theta_1)\cos(\theta_3-\theta_2) + 300\cos(\theta_2) - d_{tool}\sin(\theta_4+\theta_3-\theta_2) + 50
$$

$$
R21 = \cos(\theta_4+\theta_3-\theta_2)\sin(\theta_1)\cos(\theta_5) + \sin(\theta_1)\sin(\theta_5)
$$

$$
R22 = -\cos(\theta_4+\theta_3-\theta_2)\sin(\theta_1)\sin(\theta_5) + \cos(\theta_1)\cos(\theta_5)
$$

$$
R23 = -\sin(\theta_4+\theta_3-\theta_2)\sin(\theta_1)
$$

$$
Py = 300\sin(\theta_1)\cos(\theta_2) + 350\cos(\theta_3-\theta_2) - d_{tool}\sin(\theta_4+\theta_3-\theta_2) + 50
$$

$$
R31 = \sin(\theta_4+\theta_3-\theta_2)\cos(\theta_5)
$$

$$
R32 = -\sin(\theta_4+\theta_3-\theta_2)\sin(\theta_5)
$$

$$
R33 = \cos(\theta_4+\theta_3-\theta_2)
$$

$$
Pz = -300\sin(\theta_2) + 350\sin(\theta_3-\theta_2) - d_{tool}\cos(\theta_4+\theta_3-\theta_2) + 358.5
$$

3.2 Inverse kinematic

In the case that all components of matrix ${}^{0}_{5}T$ are known, the set of angular movements $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ can be computed via the inverse kinematics. Firstly, to find θ_1, it is estimated from below ratio

$$
\frac{Px}{Py} = \frac{\sin(\theta_1)[300\cos(\theta_2) + 350\cos(\theta_3-\theta_2) - d_{tool}\sin(\theta_4+\theta_3-\theta_2) + 50]}{\cos(\theta_1)[300\cos(\theta_2) + 350\cos(\theta_3-\theta_2) - d_{tool}\sin(\theta_4+\theta_3-\theta_2) + 50]} = \tan(\theta_1) \quad (8)
$$

Or,

$$
\theta_1 = a\tan 2(Py, Px) \quad (9)
$$

Consequently, we can make a ratio as below

$$
\frac{R32}{R31} = \frac{-\sin(\theta_4+\theta_3-\theta_2)\sin(\theta_5)}{\sin(\theta_4+\theta_3-\theta_2)\cos(\theta_5)} = -\tan(\theta_5) \quad (10)
$$

Or,

$$
\theta_5 = a\tan 2(-R32, R31) \quad (11)
$$

However, to calculate the values of $\theta_2, \theta_3, \theta_4$ some additional steps for computing should be considered.

$$R33 = \cos(\theta_4 + \theta_3 - \theta_2) \quad (12)$$

$$\theta_4 + \theta_3 - \theta_2 = a\cos(R33) \quad (13)$$

Moreover, we have

$$\cos(\theta_1)Px + \sin(\theta_1)Py = 300\cos(\theta_2) + 350\cos(\theta_3 - \theta_2) - d_{tool}\sin(\theta_4 + \theta_3 - \theta_2) + 50 \quad (14)$$

We denote

$$\theta = \theta_4 + \theta_3 - \theta_2 \quad (15)$$

$$A = \cos(\theta_1)Px + \sin(\theta_1)Py - 50 + d_{tool}\sin(\theta) \quad (16)$$

$$B = Pz - 358.5 + d_{tool}\cos(\theta) \quad (17)$$

From above equations, a set of computations is

$$\begin{cases} B = -300\sin(\theta_2) + 350\sin(\theta_3 - \theta_2) \\ A = 300\cos(\theta_2) + 350\cos(\theta_3 - \theta_2) \end{cases} \leftrightarrow \begin{cases} B + 300\sin(\theta_2) = 350\sin(\theta_3 - \theta_2) \\ A - 300\cos(\theta_2) = 350\cos(\theta_3 - \theta_2) \end{cases} \quad (18)$$

Furthermore

$$(B + 300\sin(\theta_2))^2 + (A - 300\cos(\theta_2))^2 = 350^2 \quad (19)$$

$$\leftrightarrow \frac{B^2 + A^2 + 300^2 - 350^2}{600} = A\cos(\theta_2) - B\sin(\theta_2)$$

It is assumed that the left side of above equation is C so that the value of θ_2 can be found

$$\theta_2 = a\tan 2(C; \sqrt{B^2 + A^2 - C^2}) - a\tan 2(A; -B) \quad (20)$$

Subsequently,

$$\frac{B + 300\sin(\theta_2)}{A - 300\cos(\theta_2)} = \tan(\theta_3 - \theta_2)$$

$$\theta_3 = \theta_2 + a\tan 2(B + 300\sin(\theta_2); A - 300\cos(\theta_2))$$

$$\theta_4 = \theta - \theta_3 + \theta_2 \quad (21)$$

(22)

3.3 Electric connection

Due to the specific configuration of this hardware, only four driving motors use 12V power supply while the DC 24V is applied for the others. In addition, it is necessary to separate two power supplies because of safe operation. In the host computer, an operator would handle the whole behaviors of the robot via data exchange from the camera, virtual reality headset, and MCU (microcontroller unit). As a result, the wiring diagram of our system is demonstrated in Fig. 2.

3.4 Peripheral device

Additionally, servo driver MSD E3A as Fig. 3 supports many features, for example Pulse/Direction, UART (Universal Asynchronous Receiver-Transmitter), virtual serial communication, overload protection, overcurrent protection, overheat protection. Also, it can detect broken encoder, damaged motor and so on. Since our hardware needs to control the precise position, smooth velocity and acceleration for a DC servo motor, this driver is suitable for small and medium applications that need to manipulate the angular position or speed of an actuator or robot.

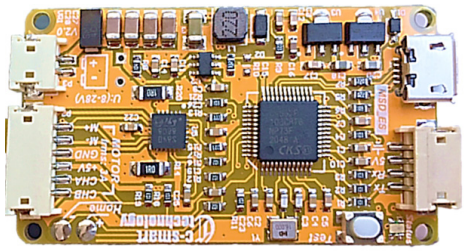


Figure 3. Description of MSD-E3 driver.

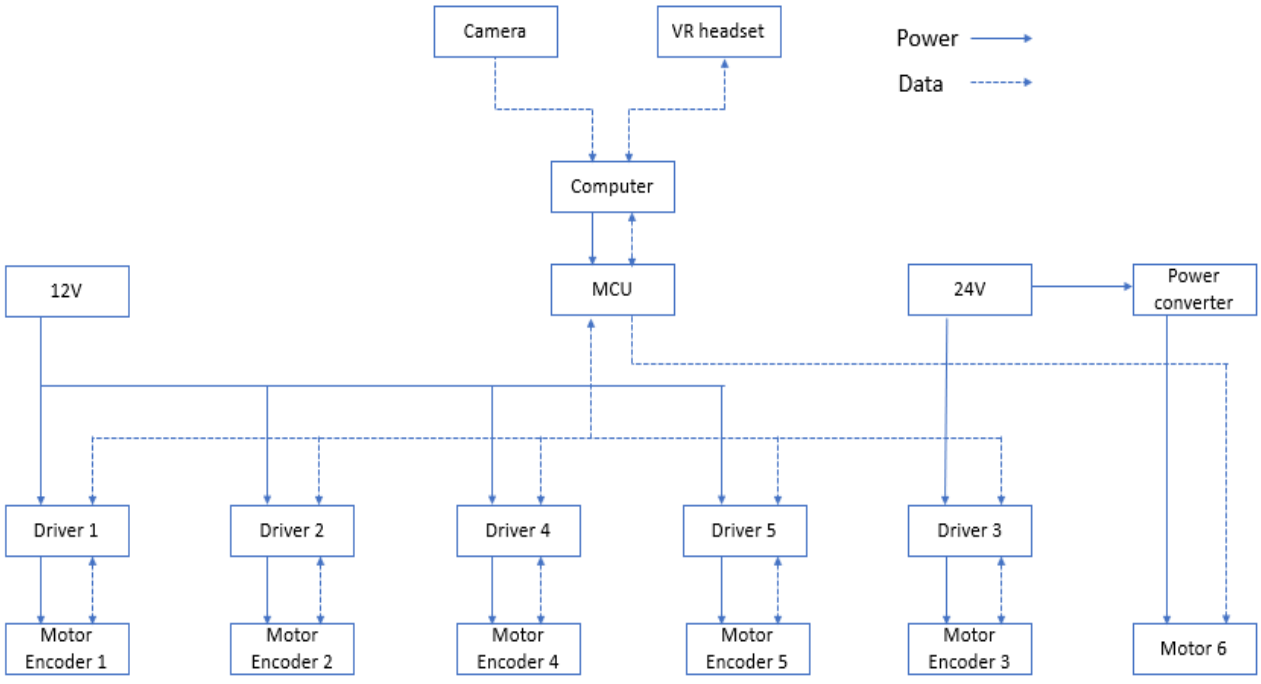


Figure 2. Wiring connection in the proposed system.

Owing to the burden computation, it is essential to explore the powerful CPU (Central Processing Unit) MSP-EXP430G2ET as Fig. 4. It has a 16 MHz core processor with 8KB Flash, 256B SRAM, 10-bit ADC, comparator, and timer. Furthermore, this CPU supports data exchange between the host computer and low-level hardware. In this case, most of the motion functions for robotic control are implemented into firmware. The compact size, low cost and wide range of applications are also advantageous for developers.

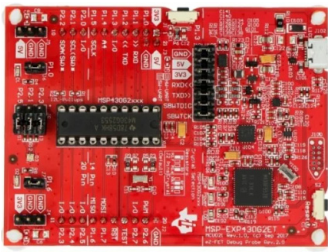


Figure 4. Description of MSP-EXP430G2ET board.

In Fig. 5, LM2596M module is utilized to cut down the voltage supply in order to provide proper power. This module is capable of adjusting the output current to 3A. LM2596 is a fully on-board IC (Integrated Circuit) inside when applying 9V power to the module. After reducing the voltage, the power source with 3A less than 9V, like 5V or 3.3V could be ensured.

Figure 5. Description of LM2596M module.

4. THE PROPOSED APPROACH

In Fig. 6, our concept is to propose the interactive platform between an operator and robotic manipulator. In the hardware side, the robot is driven by several servo drivers in the power unit. It then feedbacks the control signal to MCU which is embedded in the motion controller. An operator would use VR headset to interact with this system. In host computer, In the host computer, the control program using Visual C++ is executed to receive the visual data from the digital camera. A novel idea to embed the potential field method for enhancing the interactive performance is innovated. In addition, our motion controller is proposed to manipulate the robotic hardware according to the input signals.

4.1 Working environment setup

It is supposed that there are one table (with dimension 500mm in length, 300mm in width) and a leader tube. An operator pulls this tube to desired location, and later robotic arm would follow this trajectory. To match the coordinates of leader tube and robot, it is necessary to yield a matrix transformation by using the inverse kinematic method.

Following, the digital camera needs to capture the visual data from obstacles by integrating the OpenCV library which provides many sets of programming func–tions to mainly aim the advanced techniques of computer vision. The principal programming language in this work is C# in order to transfer data to Unity and socket communication based on local host.

In the experimental works, the sampling time which delivers the up-to-date control data to drive robot mostly depends on the processing speed of computer. After measuring the position and pose from the order of an operator, it is fused with data from camera to perform more accurate values.

4.2 The proposed algorithm

Generally, there are two components in the method of artificial potential field such repulsion potential energy and attractive potential energy. This concept derives from the fact that both potential energies fluctuate while the robot moves toward the target. When it wants to interact with an object, the attractive potential energy is greater than repulsion potential energy. Reversely, its repulsion potential energy would be larger if the robot needs to avoid it.

A. Interactive controller

In our approach, the obstacle would activate a repulsive force on the robotic arm from a specific distance as Fig. 7. As well, the force orientation would probably be opposite to the direction of the velocity vector from the robot arm to the obstacle. Since our robot has several links, these forces would be applied to each one. The repulsion potential energy of each link can be calculated as

$$Uir = \begin{cases} \frac{1}{2} Kr\left(\frac{1}{di} - \frac{1}{dsafe}\right)^2, di \le dsafe \\ 0, di > dsafe \end{cases} \quad (23)$$

where

i: the order of each link K_r: repulsion gain

d_i: distance between link to obstacle

d_{safe}: safe distance which allows robot work in normal mode

Then, total repulsion potential energy for our manipulator is

$$Urep = \sum_{i=1}^{n} \sum_{j=1}^{m} Uir \quad (24)$$

where

m: the number of obstacles having influence on a manipulator

On the other hand, it is considered that target object produces an attractive gravity to the end-effector when robot reaches to destination.

$$d_{tool} \quad (25)$$

where

K_a: attractive gain

X_{goal}: coordinate of the target

X_l: coordinate of the end-effector

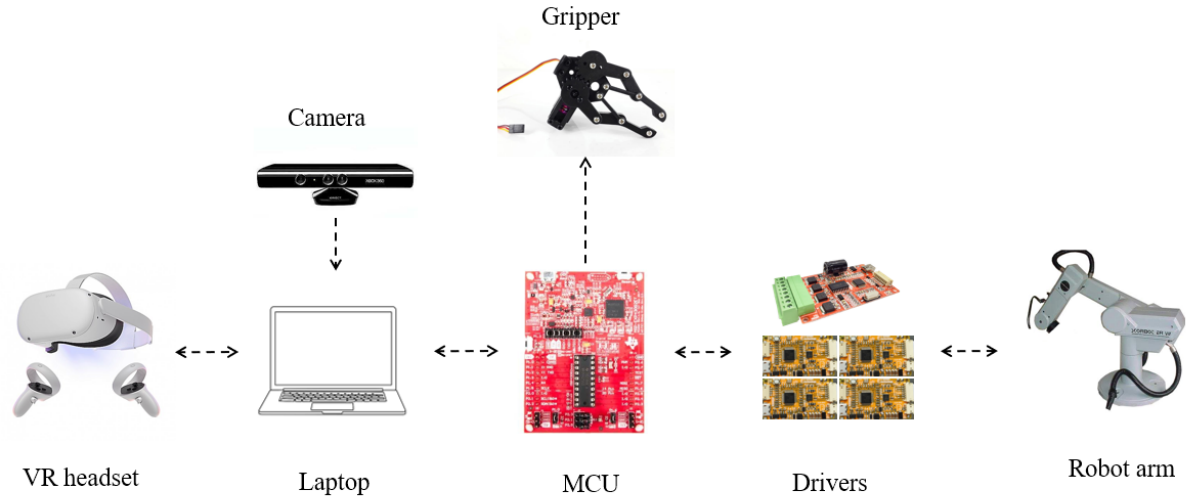


Figure 6. Overall scheme of the proposed system.

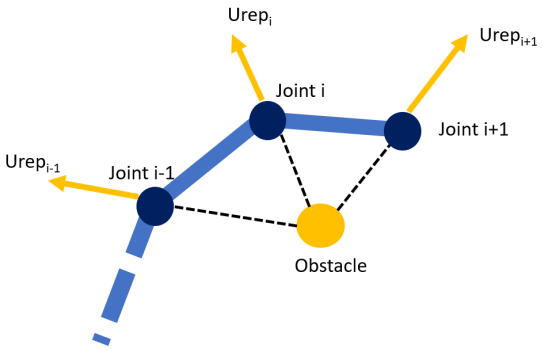


Figure 7. Description of the repulsive energy on each link while obstacle stays in the working space.

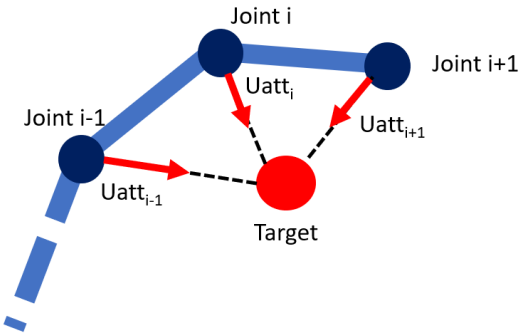


Figure 8. Description of the attractive energy between the end-effector and target object.

Last but not least, the summation of both repulsive energy and attractive energy performs the artificial potential field which is useful to illustrate the interaction between robotic platform and an object. This field could be expressed as

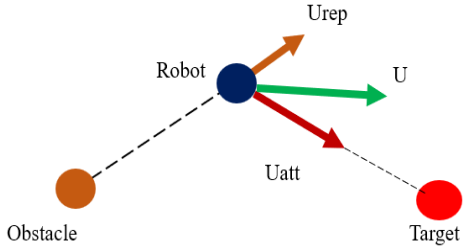


Figure 9. Description of the artificial potential field in the interaction between robot and an object.

B. Motion controller

$$U = Uatt + Urep \tag{26}$$

where

Urep: Repulsion potential energy

Uatt: Attractive potential energy

We assume that the driver-motor system is linear in separated link. Thus, the PI (Proportional-Integral) control rule is proper to handle the driving motor as Fig. 10. The mathematical expression of this scheme can be described as below:

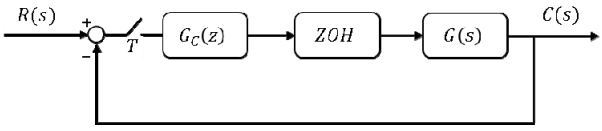


Figure 10. Description of the PID control scheme.

where,

$R(s)$: Input signal

$C(s)$: Output signal

$G_c(z)$: Transfer function of discrete system

$G(s)$: Transfer function of driver-motor system

ZOH: Zero-order hold function block

$$G_c(z) = K_P + K_I T \frac{z}{z-1} \tag{27}$$

Due to our experiences, the control parameters for this system should be determined as following: steady-state time Ts < 0.1(s), percentage of overshoot $\%\frac{OS}{100} < 10\%$, steady-state error $ess < 5(s)$. The boundary value of overshoot such as 10% can be inserted to compute the damping ratio ζ.

$$\zeta = \frac{-\ln(\%\frac{OS}{100})}{\sqrt{\pi^2 + \ln^2(\%\frac{OS}{100})}} \tag{28}$$

Hence, natural frequency ω_n is

$$\omega_n = \frac{4}{\zeta T_s} \tag{29}$$

The poles of system are required as

$$z_1^* = re^{+j\varphi} \tag{30}$$

$$z_2^* = re^{-j\varphi} \tag{31}$$

where,

$$r = e^{-T\zeta\omega_n}$$

$$\varphi = T\omega_n\sqrt{1-\zeta^2}$$

As a result, transfer function of the open-loop system is identified as

$$G(z) = (1-z^{-1})Z\left\{\frac{G(s)}{s}\right\} \tag{32}$$

5. RESULT OF STUDY

To verify the practicality and usefulness of our method, several simulations and experiments are suggested to establish. Owing to our design of the wiring diagram, the electrical cabin is shown in Fig. 11. With the dimension 500x450x250, it is chosen to minimize the wiring area such that four plastic panel boards are fixed in the cabin to hold electrical components. By using this scheme, the safe condition in respect to the reversed current can be confirmed.

There are several VR headsets in the commercial market, i.e. oculus, VR gear or HTC. They are supported by many mobile apps and frameworks. Largely, all functions that need for this study, are included in these items. However, some of them have negative aspects since the device can not track position in space while computer software does not support to combine with the mobile apps on headset. Hence, the oculus presented in Fig. 11 becomes the best choice because of the lower price and fundamental functions.

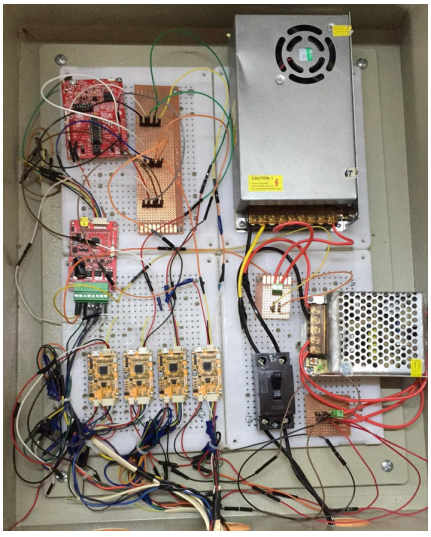


Figure 11. Practical wiring connection of the electrical cabin.

Moreover, our validations are completed in host computer with core i7 3.7GHz, RAM 512 GB, OS Win 11. To obtain the results of numerical simulations, all tests are done in Matlab/Simulink with the same setting parameters as our design. In the experimental verifications, a virtual model of our manipulator could be launched in Unity 3D.

Figure 12. Description of Oculus headset.

5.1 Simulation result

To start the 2D simulation, several initial values of parameters must be set such as attractive gain (kp=1), repulsive gain (kr= 2), initial position of the end effector (x = 468, y = 0, z = 313), 1st desired position of the end-effector (x = 468, y = 500, z = 313), 2nd desired position of the end-effector (x = 468, y = 0, z = 313), control step of trajectory (y = y + 10), position of the obstacle (x = 468, y = 250, z = 313), safe distance (dsafe = 100mm).

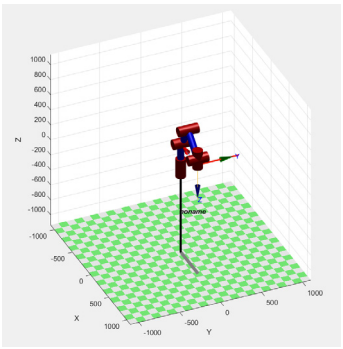
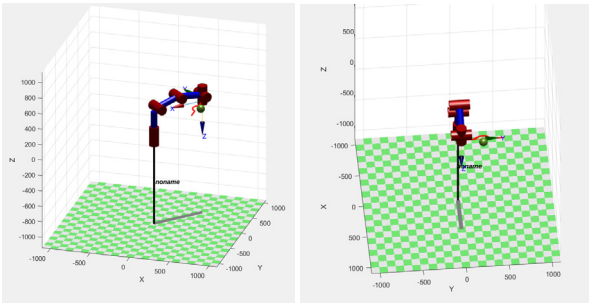


Figure 13. Simulation result for free motion in 2D environment using our system.



(a) (b)

Figure 14. Simulation result for obstacle avoidance in 2D environment using our system, (a) at initial position and (b) at final position.

In Fig. 13, it is shown that the directional movement of end-effector could be ensured. In this case, there are approximately 360 test points with the radius 1mm. More test points are set, smaller moving radius must be obligated to track highly accurate points.

Though, the time consuming would be longer when the number of points increases. Henceforward, the tuning gains must be found so that this system tries to catch the

desired trajectory. In Fig. 14, owing to the step angle, robot can travel smoothly and stably to overcome this obstacle.

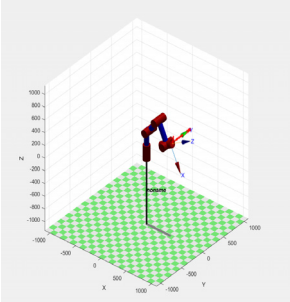
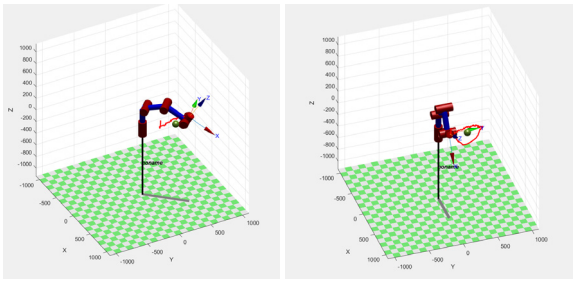


Figure 15. Simulation result for free motion in 3D environment using our system.

In the 3D simulation, manipulator must carry out more flexible motion and generate more complicated profile. In Fig. 15, the system parameters at initial stage are the same in order to measure the behaviour of robot in different contexts. From Fig. 16, it can be seen clearly that our system can adapt to various working conditions flexibly and stably.

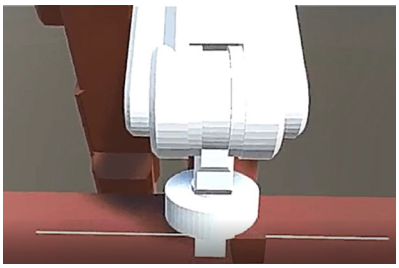


(a) (b)

Figure 16. Simulation result for obstacle avoidance in 3D environment using our system, (a) at initial position and (b) at final position.

5.2 Experimental result

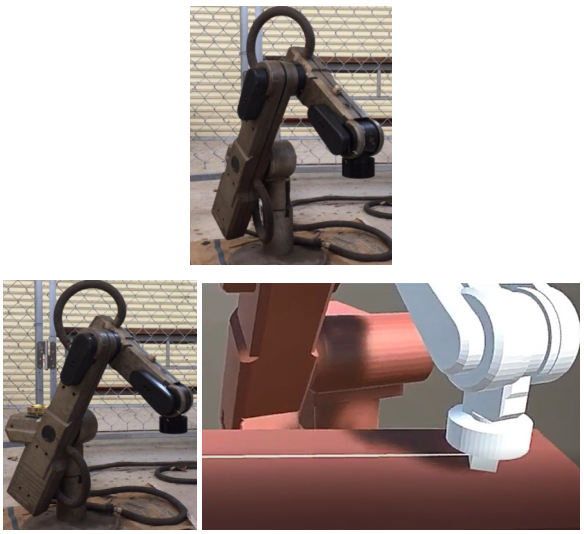
In this section, the proposed system has been verified by several tests. Regularly, the starting location of robotic manipulator should be homing position. Later, its trajectory which is generated by our algorithm, imitates the theoretical planning path in order to evaluate the practical performance. Furthermore, due to some limitations in physical hardware such as noise or unexpected factors, this manipulator needs additional filters to guarantee the proper signals. In Fig. 17, robot is originally set at the start position.



(a) (b)

Figure 17. Experimental result for the synchronous motion between our robot (a) and its virtual model (b) at initial position.

Also, the virtual model of the robot is located in the same condition to compare the working configuration. In front of end-effector, it is necessary to create an obstacle which is only appeared in the simulated environment. In our experiment, the linear trajectory is planned for the robotic manipulator to track. Although there exist some diffe–rences between physical platform and cyber model, the proposed system tries to follow the desired paths as Fig. 18.



(a) (b)

Figure 18. Experimental result for the synchronous motion between our robot (a) and its virtual model (b) at final position.

To measure the practical performance of driving mechanism, several examinations should be carried out. Since there are the same types of servo motors, one of them could be verified as an example. Using motion controller as mentioned above, transfer function of driving motor could be calculated as following

$$\begin{aligned} G(z) &= (1-z^{-1})Z\left\{\frac{958.6}{s(s+51.49)}\right\} \\ &= (1-z^{-1})\frac{958.6z(1-e^{-0.01\times 51.49})}{51.49(z-1)(z-e^{-0.01\times 51.49})} \\ &= \frac{958.6(1-e^{-0.01\times 51.49})}{51.49(z-e^{-0.01\times 51.49})} \\ &= \frac{7.49}{z-0.6} \end{aligned}$$

Later, the system equation which is characterized by its performance, is

$$1+G_C(z)G_L(z)=0$$

$$\Leftrightarrow 1+(K_P+K_IT\frac{z}{z-1})\left(\frac{7.49}{z-0.6}\right)=0$$

$$\Leftrightarrow (z-1)(z-6)+7.49K_P(z-1)+0.07K_Iz=0$$

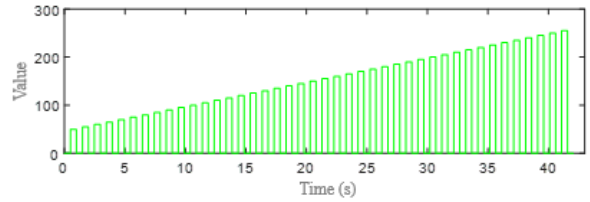
$$\Leftrightarrow z^2-1.6z+0.6+7.49K_Pz-7.49K_P+0.07K_Iz=0$$

$$\Leftrightarrow z^2+(7.49K_P+0.07K_I-1.6)z+(-7.49K_P+0.6)=0$$

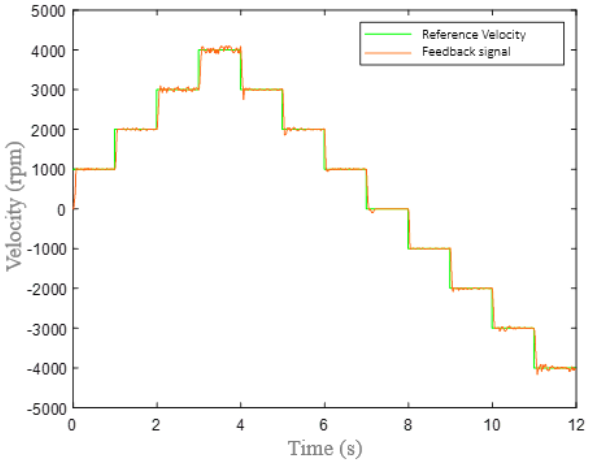
Based on the desired poles, the system equation should be

$$(z-0.57-0.35j)(z-0.57+0.35j)=0$$

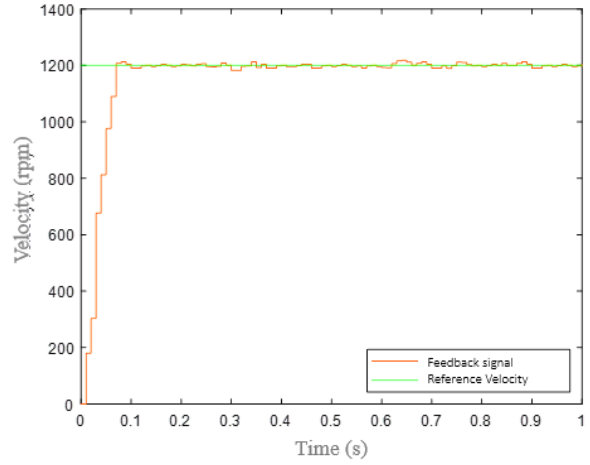
$$\Leftrightarrow z^2 - 2\times 0.57z + 0.57^2 + 0.35^2 = 0$$
$$\Leftrightarrow z^2 - 1.15z + 0.45 = 0$$



(a)



(b)



(c)

Figure 19. Experimental result for the PWM signal of the driving motor (a), continuous variations in speed (b) and tracking performance (c).

From above computations, the control gains could be estimated as following

$$\begin{cases} 7.49K_P + 0.07K_I - 1.6 = -1.15 \\ -7.49K_P + 0.6 = 0.45 \end{cases} \Leftrightarrow \begin{cases} K_P^L = 0.02003 \\ K_I^L = 4.28571 \end{cases}$$

In the real-world experiment, a set of control parameters can be chosen as $K_P^L = 0.02$, $K_I^L = 2.3$. In each link, the desired linear velocity and angular velocity can be obtained via inverse kinematic. Then, these values are transmitted to main processor for driving commands. However, in the level of hardware, it is necessary to evaluate the output signals of our controller. Therefore, data from one driving motor is sampled for visualizing the motion control process. In Fig. 19, several experimental results have been completed to verify the system response. According to those achievements, it can be seen clearly that the proposed design is proper and feasible for various applications of cyber-physical system.

To compare the specific characteristics and features among related fields, Table 3 depicts the competitive performance in platform, working environment, type of reality and technique. In [28], developers used Hololens glasses to interact with an operator. Depending on the robotic configuration, trajectory planner was generated by 3D point-cloud from digital camera. However, their drawbacks are to be difficult to precisely estimate via those point-clouds, too complicated all objects on the screen and limited users in the same application. In the other approach [29], virtual barrier was suggested as a safe solution in manufacturing. In that case, robot would recognize each zone such as task area, material piece and replacement area. Then, robot could manipulate with object or human in respect to their hypothesis. Though, less intuitive methods of interacting with virtual objects and simple geometries in trial objects are possibly challenges of this method.

Table 3. List of the competitive specifications in related works.

Author(s)	Robot platform	Virtual environment	Type	Technique
Ostanin, M. et al [28]	UR10e, KUKA iiwa	ROS (Robotic Operating System)	Mixed Reality	3D Point-cloud
Hoang, K. C. et al [29]	N/A	ROS (Robotic Operating System)	Augmented Reality	Virtual barrier by way-points
Our approach	SCORBOT-ER-IX	Unity 3D	Augmented Reality	Potential field

N/A: Not applied

From above study, it is potentially indicated that the proposed approach could be deployed in teleoperation or remote control of automated system, training and maintenance, collaboration and task assistance or social interaction, safety and monitoring. To support the remotely robotic manipulation, AR can enable intuitive gesture-based controls, and operators manage hardware platform with additional contextual information. Besides, this approach can create realistic simulations for training purposes, allowing users to practice interacting with robots in a safe, controlled environment. Furthermore, to share task context, our method can highlight tools and objects involved in collaborative activities, and assist in planning by displaying virtual guides. For more practices, this method can provide real-time monitoring of robot performance and working conditions, alerts potential hazards in the environment.

6. CONCLUSION

In this investigation, a novel idea to design a framework for communication between cyber model and physical hardware via augmented reality (AR) technique is

developed. The AR technique is embedded into the closed control loop in order to provide the interactive effects from an operator. With this design, both simulation and experiment have been validated. From these results, some potential developments are encouraged. Virtual objects which cost much time to produce, should be added for robotic manipulation. These objects could be placed on the body of robot or separately. Moreover, the artificial intelligence technique and vision-based method could be applied to improve its behaviours.

Although our efforts were made, several barriers still exist in this study. It lacks of deeply analysing in time consumption as well as data latency in communication. Furthermore, some machine learning schemes could be deployed to predict the system response in both cyber model and physical platform. In further developments, above challenges should be addressed.

ACKNOWLEDGMENT

We acknowledge Ho Chi Minh City University of Tec–hnology, (VNU-HCM) for supporting this study.

CONFLICT OF INTEREST

Authors have no conflict of interest relevant to this article.

REFERENCES

[1] Dianatfar, M., Latokartano, J., & Lanz, M. (2021). Review on existing VR/AR solutions in human–robot collaboration. Procedia CIRP, 97, 407-411.

[2] Suzuki, R., Karim, A., Xia, T., Hedayati, H., & Marquardt, N. (2022, April). Augmented reality and robotics: A survey and taxonomy for ar-enhanced human-robot interaction and robotic interfaces. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (pp. 1-33).

[3] Makhataeva, Z., & Varol, H. A. (2020). Augmented reality for robotics: A review. Robotics, 9(2), 21.

[4] Mara, M., Stein, J. P., Latoschik, M. E., Lugrin, B., Schreiner, C., Hostettler, R., & Appel, M. (2021). User responses to a humanoid robot observed in real life, virtual reality, 3D and 2D. Frontiers in psychology, 12, 633178.

[5] Cong, V. D. (2024). Design and Development of a Costefficiency Robot Arm with a PLC-based Robot Controller. FME Transactions, 52(2).

[6] Danielsson, O., Holm, M., Syberfeldt, A. (2020). Augmented reality smart glasses in industrial as–sembly: Current status and future challenges, Journal of Industrial Information Integration, 20, 100175.

[7] Kent, L., Snider, C., Gopsill, J., & Hicks, B. (2021). Mixed reality in design prototyping: A systematic review. Design Studies, 77, 101046.

[8] Pratticò, F. G., Lamberti, F. (2020). Mixed-reality robotic games: design guidelines for effective entertainment with consumer robots. IEEE Consumer Electronics Magazine, 10(1), 6-16.

[9] Kim, H., Everitt, A., Tejada, C., Zhong, M., Ashbrook, D. (2021, May). Morpheesplug: A toolkit for prototyping shape-changing interfaces. In Pro–ceedings of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-13).

[10] Nguyen, T. T., Nguyen, T. H., Ngo, H. Q. T. (2024). Investigation on the Mechanical Design of Robot Gripper for Intelligent Control Using the Low-cost Sensor. FME Transactions, 52(1).

[11] Nguyen, T. V., Lê, L. S., Shah, S. A., Hameed, S., Draheim, D. (2023). PenChain: A Blockchain-Based Platform for Penalty-Aware Service Provision–ing. IEEE Access.

[12] Ngo, H. Q. T., Nguyen, H., & Nguyen, T. P. (2023). Fenceless collision free avoidance driven by visual computation for an intelligent cyber physical system employing both single and double S trajectory. IEEE Transactions on Consumer Electronics.

[13] Ngo, H. Q. T. (2023). Using an HSV-based App–roach for Detecting and Grasping an Object by the Industrial Manipulator System. FME Transactions, 51(4).

[14] Nguyen, T. P., Nguyen, H., Ngo, H. Q. T. (2023). Visual application of navigation framework in cyber-physical system for mobile robot to prevent disease. International Journal of Advanced Robotic Systems, vol. 20, no. 2, https://doi.org/10.1177/17298806231162202, 2023.

[15] Walker, M., Hedayati, H., Lee, J., & Szafir, D. (2018, February). Communicating robot motion intent with augmented reality. In Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction (pp. 316-324).

[16] Pjanic, P., Willi, S., Grundhöfer, A. (2017). Geo-metric and photometric consistency in a mixed video and galvanoscopic scanning laser projection mapping system. IEEE Transactions on Visualization and Computer Graphics, 23(11), 2430-2439.

[17] Nguyen, T. P., Nguyen, H., Ngo, H. Q. T. (2023). Developing and evaluating the context-aware per–formance of synchronization control in the real-time network protocol for the connected vehicle. Mobile Networks and Applications, 1-22.

[18] Nguyen, T. P., Nguyen, H., Thinh Ngo, H. Q. (2022). Planning the Emergency Collision Avoidance Strategy Based on Personal Zones for Safe Human-Machine Interaction in Smart Cyber-Physical System. Complexity, vol. 2022, 1-21, https://doi.org/10.1155/2022/2992379, 2022.

[19] Arlotti, J. S., et al. (2022). Benefits of IMU-based wearables in sports medicine: Narrative review, International Journal of Kinesiology and Sports Science, 10(1), 36-43.

[20] Adilkhanov, A., Rubagotti, M., Kappassov, Z. (2022) Haptic devices: Wearability-based taxonomy and literature review. IEEE Access, 10, 91923-91947.

[21] Cacace, J., Finzi, A., Lippiello, V. (2017, August). A robust multimodal fusion framework for command interpretation in human-robot cooperation. In 2017 26th IEEE International Symposium on Robot and

Human Interactive Communication (ROMAN) (pp. 372-377). IEEE.

[22] Angleraud, A., Mehman Sefat, A., Netzev, M., Pieters, R. (2021). Coordinating shared tasks in human-robot collaboration by commands. Frontiers in Robotics and AI, 8, 734548.

[23] Korayem, M. H., Azargoshasb, S., Korayem, A. H., Tabibian, S. (2021). Design and implementation of the voice command recognition and the sound source localization system for human–robot interaction. Robotica, 39(10), 1779-1790.

[24] Dianatfar, M., Latokartano, J., Lanz, M. (2020). Concept for virtual safety training system for human-robot collaboration. Procedia Manufactu–ring, 51, 54-60.

[25] Guzzi, J., Abbate, G., Paolillo, A., Giusti, A. (2022, March). Interacting with a conveyor belt in virtual reality using pointing gestures. In 2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI) (pp. 1194-1195). IEEE.

[26] Yongda, D., Fang, L., Huang, X. (2018). Research on multimodal human-robot interaction based on speech and gesture. Computers & Electrical Engineering, 72, 443-454.

[27] Malik, A. A., Masood, T., Bilberg, A. (2020). Vir–tual reality in manufacturing: immersive and col–laborative artificial-reality in design of human-robot workspace. International Journal of Computer Integrated Manufacturing, 33(1), 22-37.

[28] Ostanin, M., Mikhel, S., Evlampiev, A., Skvor–tsova, V., Klimchik, A. (2020, May). Human-robot interaction for robotic manipulator programming in Mixed Reality. In 2020 IEEE International Conference on Robotics and Automation (ICRA) (pp. 2805-2811). IEEE.

[29] Hoang, K. C., Chan, W. P., Lay, S., Cosgun, A., Croft, E. (2022, August). Virtual barriers in aug-mented reality for safe and effective human-robot cooperation in manufacturing. In 2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) (pp. 1174-1180). IEEE.

NOMENCLATURE

x_i, y_i, z_i	local coordinate of link ith
a_i	the distance between z_{i-1} axis and z_i axis along the x_i –axis
α_i	the required rotation of the z_{i-1}-axis about the x_i -axis to parallel the z_i –axis
d_i	the distance between x_{i-1} axis and x_i axis along the z_{i-1}-axis
θ_i	the required rotation of x_{i-1} -axis about the z_{i-1}-axis to parallel the x_i –axis
d_{tool}	the distance from the center of the 4th coordinate to the center of the end-effector
${}^{i-1}_{i}T$	matrix to transform coordinate frames B_i to B_{i-1}
K_r	Repulsion gain
d_i	distance between link to obstacle
d_{safe}	safe distance which allows robot work in normal mode
Uir	Repulsion potential energy
K_a	Attractive gain
X_{goal}	Coordinate of the target
X_1	Coordinate of the end-effector
$R(s)$	Input signal
$C(s)$	Output signal
$G_c(z)$	Transfer function of discrete system
$G(s)$	Transfer function of driver-motor system
ZOH	Zero-order hold function block

ABBREVIATIONS

AR	Augmented Reality
DOFs	Degree Of Freedoms
D-H	Denavit-Hartenberg
UART	Universal Asynchronous Receiver-Transmitter
CPU	Central Processing Unit
MCU	Micro Controller Unit
IC	Integrated Circuit
PI	Proportional-Integral

ДИЗАЈН ОКВИРА КОРИШЋЕЊЕМ РОБОТСКЕ ПРОШИРЕНЕ СТВАРНОСТИ ЗА САЈБЕР-ФИЗИЧКИ СИСТЕМ

Л.Ф. Нгујен, Х.К.Т. Нго

У контексту Индустрије 4.0, многе напредне технологије су истражене како би се побољшале високе перформансе тренутно роботског система. У овом раду представљен је нови концепт комуникације између сајбер модела и физичког хардвера путем технике проширене стварности (АР). Што се тиче хардверске конфигурације, описане су и предња и инверзна кинематика да би се робот правилно контролисао. Поред тога, укратко су приказане неке компоненте практичног хардвера. За сајбер оквир, контролни алгоритам као и софтвер за симулацију су успостављени у виртуелном окружењу. Да би се потврдио предложени дизајн, сајбер модел и физички хардвер који користе наш оквир су симулирани и експериментисани. Из ових резултата се може јасно видети да је наш приступ ефикасан и изводљив за практичну примену.