

Automatic Dialing Test System Based on Visual Servo Control

Kuan-Yu Chou and Jih-Gau Juang*

Department of Communications, Navigation and Control Engineering, National Taiwan Ocean University,
2 Pei-Ning Road, Keelung 20224, Taiwan

(Received April 22, 2020; accepted April 22, 2021)

Keywords: robot arm, D-H model, visual servo, image processing, fuzzy control

In recent years, with the progress of Industry 4.0, intelligent robotic technologies have been widely used in various fields, such as assembly on a production line and performing minimally invasive surgery. We propose the application of a five-degree-of-freedom robot arm to the automatic testing of smartphones. A camera is set up on a robot arm and the images captured by the camera are used for visual servo control of the robot arm. Image processing is performed by MATLAB, which first converts the red–green–blue (RGB) color space into the hue–saturation–value (HSV) color space, then binarizes the image, and then filters out noises. The control system applies LabVIEW to integrate the image and fuzzy control for the angles of each joint in the movement of the robot arm. By converting the coordinates through the Denavit–Hartenberg (D–H) model, the degrees of joints can be updated by visual servoing without the need for camera calibration. Fuzzy theory is used to obtain the error of the robot arm and then send the command to adjust the movement of the robot arm. Experimental results show that the proposed control system can drive the robot arm to press the required buttons of test smartphones.

1. Introduction

To respond to Industry 4.0, this study mainly focuses on improving the efficiency of a smartphone test program to replace labor and provide more applications for robotic arms. Intelligent robot control has become more common in recent years, and the technologies include image processing, path planning, and arm control.^(1–6) In previous works,⁽⁷⁾ mobile phone tests were performed using a fixed camera, so that mistakes had to be canceled and reprinted. The main purpose of this study is to increase the accuracy of clicking buttons by means of visual servo control.^(8–11) The method utilized for coordinate transformation so that the robot arm can be driven to target positions is the Denavit–Hartenberg (D–H) model.⁽¹²⁾ The target positions are on the smartphone, which are the desired buttons to be pressed. Then image processing and recognition methods are used to identify the location of the target button. In the image processing, the hue–saturation–value (HSV) color space is used to convert an image taken from a webcam to a binary image. It is planned that the edge click can be corrected through the visual servo. The use of fuzzy control eliminates the need to use complicated mathematical formulas,

*Corresponding author: e-mail: jgjuang@ntou.edu.tw
<https://doi.org/10.18494/SAM.2021.3362>

which simplifies the calculation in the control process and improves the overall efficiency of the robot arm control.

The first robot arm that was specifically used for industrial production was invented by Joseph Engberg, who was called the “father of the robotic arm”. In 1956, using the patented technology authorized by George Divor,⁽¹³⁾ he developed the first industrial robotic arm called Unimate. The control methods behind robot arms are diverse and include traditional proportional-integral-derivative (PID) control, adaptive control,⁽¹⁴⁾ and fuzzy control. Because a robot is composed of many rods and nodes, its purpose is to drive the arm. In 1900, Mozzi and Chasles proposed screw theory, which can be used to enhance the velocity control of the robotic arm.^(15,16) Then, in 1955, Denavit and Hartenberg proposed the D–H model or D–H conversion.⁽¹⁷⁾ D–H conversion can be performed simultaneously in column coordinates, Cartesian coordinates, and spherical coordinates. Because the fixed axis does not use the Y -axis, letting the rotary joint follow the right-hand rule determines the Z -axis, and the translational direction determines the X -axis, which can reduce the complexity of the coordinate conversion. Therefore, D-H conversion has been used on robot arms.^(17–19)

Machine vision involves a detection machine equipped with a sensing visual instrument (such as an autofocus camera or sensor), in which the optical detection instrument occupies a very high proportion of the space and can be used to detect defects in various products or to judge and select objects, measure the size of objects, and so forth. It has been applied to automated production lines to calibrate and position materials. It is the most industrialized part of computer vision and is mainly used in factory automation testing and robotics. Visual servos began to appear as early as the 1990s. Hutchinson *et al.* proposed image feedback control in 1996,⁽²⁰⁾ which is divided into an image-based visual servo (IBVS) and a position-based visual servo (PBVS). The former deals with the image of the target object, while the latter handles the position of the object in 3D coordinates. The camera used in a visual servo can be placed directly at a fixed position or placed at the end of an arm, called an “eye in hand” camera structure.⁽²¹⁾ The two architectures can be referred to as having fixed camera coordinates and dynamic camera coordinates, respectively. The camera coordinates used in this study are “eye in hand” dynamic camera coordinates. Our study is based on an IBVS system, which is combined with a mechanical arm to perform smartphone dialing tests.

2. System Description

In this study, the embedded computer is used as the main control processor. The Dynamixel MX–28 servo motor, which is produced by ROBOTIS Company, provides the main force to the robot arm, as shown in Fig. 1(a). The motor’s weight is 72 g, its dimensions are 35.6 (mm) × 50.6 (mm) × 35.5 (mm), its resolution is 0.088°, its running degree is 0–360°, its gear reduction ratio is 193:1, its stall torque is 24 kgf.cm (at 12 V, 1.5 A), and its no-load speed is 54 rpm (at 12 V). We use the Microsoft LifeCam Studio 1080P Full-HD network camera for image processing, as shown in Fig. 1(c).

The D–H model is a standard model used in determining the variables of a robot arm for operational analysis. This model is used to find values of the parameters θ_i , α_i , a_i , and d_i , which

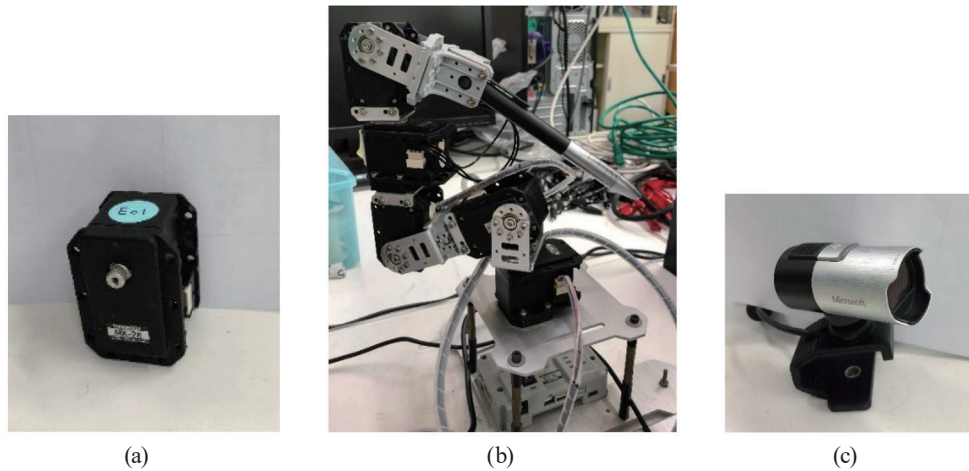


Fig. 1. (Color online) (a) Dynamixel MX-28 servo motor, (b) robot arm, and (c) Microsoft LifeCam.

are the angle about the previous z -axis from the old x -axis to the new x -axis, the angle about the common normal from the old z -axis to the new z -axis, the length of the common normal, and the offset along the previous z -axis to the common normal, respectively. The parameters are derived from the D–H model, which are obtained from the robot arm analyzed in Fig. 2. We can analyze the relationship between each joint, as shown in Table 1.

Using the D–H model, we can obtain the following homogeneous transformation matrices of a humanoid robot arm, where $c_i = \cos\theta_i$ and $s_i = \sin\theta_i$.

$$\begin{aligned}
 \mathbf{T}_1^0 &= \text{Rot}(Z, \theta_1) \times \text{Trans}(a_1, 0, 0) \\
 &= \begin{bmatrix} C_1 & -S_1 & 0 & a_1 c_1 \\ s_1 & c_1 & 0 & a_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)
 \end{aligned}$$

$$\begin{aligned}
 \mathbf{T}_2^1 &= \text{Rot}(Z, \theta_2) \times \text{Trans}(a_2, 0, 0) \\
 &= \begin{bmatrix} C_2 & -S_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)
 \end{aligned}$$

$$\begin{aligned}
 \mathbf{T}_3^2 &= \text{Rot}(Z, \theta_3) \times \text{Trans}(a_3, 0, 0) \\
 &= \begin{bmatrix} C_3 & -S_3 & 0 & a_3 c_3 \\ s_3 & c_3 & 0 & a_3 s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)
 \end{aligned}$$

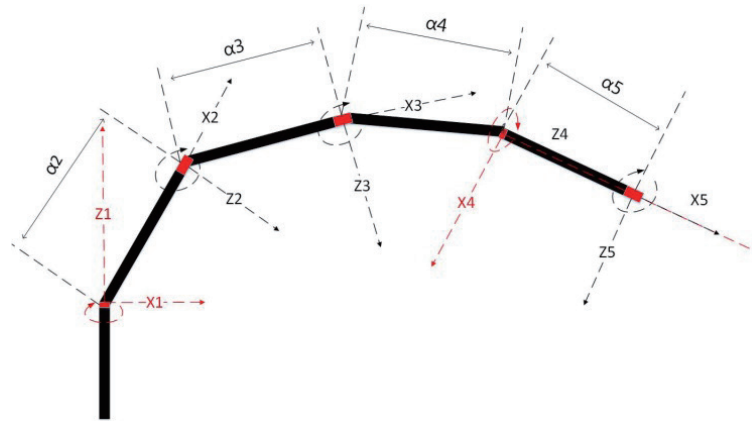


Fig. 2. (Color online) Five-degrees-of-freedom articulated robot arm analyzed using D–H model.

Table 1
D–H parameters.

Joint	θ_i	α_i	a_i	d_i
1	θ_1	90°	0	0
2	θ_2	0°	a_2	0
3	θ_3	0°	a_3	0
4	θ_4	90°	a_4	0
5	θ_5	0°	a_5	0

$$\begin{aligned}
 T_4^3 &= Rot(Z, \theta_4) \times Trans(a_4, 0, 0) \\
 &= \begin{bmatrix} C_4 & -S_4 & 0 & a_4 c_4 \\ s_4 & c_4 & 0 & a_4 s_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)
 \end{aligned}$$

$$\begin{aligned}
 T_5^4 &= Rot(Z, \theta_5) \times Trans(a_5, 0, 0) \\
 &= \begin{bmatrix} C_5 & -S_5 & 0 & a_5 c_5 \\ s_5 & c_5 & 0 & a_5 s_5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)
 \end{aligned}$$

By substituting the link parameters from Table 1 into Eqs. (1)–(4), the pose of the end effector can be obtained, where (n_x, n_y, n_z) is the orthogonal vector, (o_x, o_y, o_z) is the orientation vector, (a_x, a_y, a_z) is the approach vector, and (P_x, P_y, P_z) is the end-effector position vector. Through Eq. (6), any position in the end-effector frame can be mapped to the base frame.⁽¹²⁾

$$\begin{aligned}
 T_5^0 &= T_1^0 \times T_2^1 \times T_3^2 \times T_4^3 \times T_5^4 \\
 &= \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)
 \end{aligned}$$

3. Image Recognition

The purpose of this study is to control the robot arm to successfully reach the target position. This study uses a traditional visual servo system to capture feature points. MATLAB is used to perform color space conversion, grayscale processing, thresholding, binarization, and corner detection. Finally, the pattern is used to identify the number on a smartphone. After the identification, the coordinates are sent back to LabVIEW to control the position of the arm. By using the images obtained by the webcam, an image processing module is applied to recognize the words and symbols on a PC screen or smartphone. This process involves three main steps: image preprocessing, image recognition, and referencing a dictionary. The color space defines the color's coordinate system. All devices that capture or render colors have a color space. The color space is based on three or more color points and white points in the standard reference space. Many of the color spaces we deal with are based on red, green, and blue; we call these red–green–blue (RGB) color spaces. Another color space is the HSV color space.⁽²²⁾ The HSV model defines the color space based on the hue (H), saturation (S), and value (V) components.

The HSV color space is used in this study. Hue (H) is the basic attribute of color, which is usually called the color name, such as red, yellow, and so on. Saturation (S) refers to the purity of the color and takes a value between 0 and 100%. The higher the saturation, the purer the color, and the lower the value, the grayer it appears. Value (V) refers to the relative lightness or darkness of a color. For an image, identifying the corresponding color is possible in the RGB space, HSV space, or another color space. The reason why HSV is chosen is because the H-tone can basically determine a certain color, and then the saturation value and the lightness information can be judged when the hue is above a certain threshold. An RGB image of the test environment is taken with a network camera, then the image is converted to the HSV color space. This study uses the HSV color space as the basis for image operators. A picture taken in the RGB color space is shown in Fig. 3, and the picture after conversion to the HSV color space is shown in Fig. 4.

Recognition technology mainly includes pattern matching and character recognition. The PC screen only shows commands from different characters (numbers or words) of the control system, and we can see a strong contrast between the characters in the background. The smartphone screen also shows a similar phenomenon, but we need to consider whether the characters are in strong contrast to the background. The original image is shown in Fig. 5(a), and the binarized image is shown in Fig. 5(b).

The *match pattern* is used to quickly find a known reference in an image, and the pattern is not affected by position, direction, lightness, and temperature changes. We can create a sample

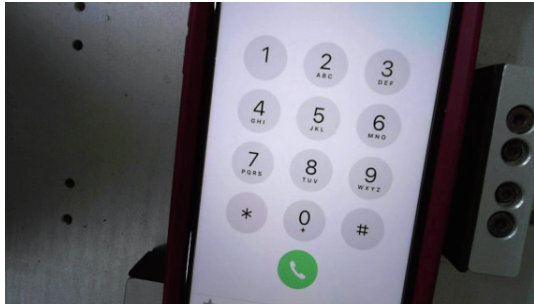


Fig. 3. (Color online) RGB image of a smartphone.

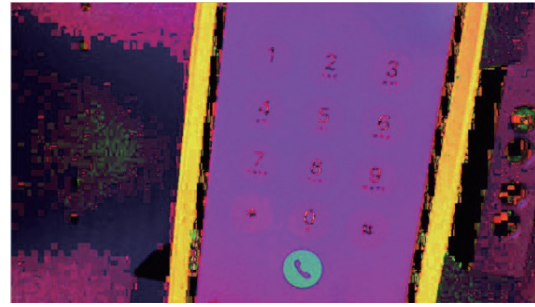


Fig. 4. (Color online) HSV image of a smartphone.



(a)



(b)

Fig. 5. (a) Image of original numbers and (b) binarized image.

model of the object we want to search for, then find the sample and calculate the similarity of each image. This model is called a template and should be an ideal representation of a pattern or object. If the similarity measure is sufficiently large, we can assume that the object exists. A cross-correlation function based on the similarity measure can be calculated. The similarity measure is based on the Euclidean distance⁽²³⁾ and is given as follows.⁽²⁴⁾ $I(x, y)$ is a common measure used when comparing the similarity of two images [e.g., a template $p(x, y)$ and test image $f(x, y)$] and is given by Eq. (7). The similarity is based on the measure of the standard Euclidean distance between two sectors, as shown in Fig. 6,⁽²⁴⁾ where L is the horizontal length of the template image, K is the vertical length of the template image, M is the horizontal axis of the test image, and N is the vertical axis of the test image.

$$I(x, y) = \sqrt{\sum_0^{L-1} \sum_0^{K-1} (f(x, y) - p(x, y))^2} \tag{7}$$

$$I(x, y)^2 = \sum_0^{L-1} \sum_0^{K-1} (f(x, y)^2 - 2 \times f(x, y) \times p(x, y) + p(x, y)^2) \tag{8}$$

The normalized cross-correlation (NCC) is used to find the incidence of patterns or objects within an image. The NCC uses the product in Eq. (9) and is scaled so that it is in the range of 0

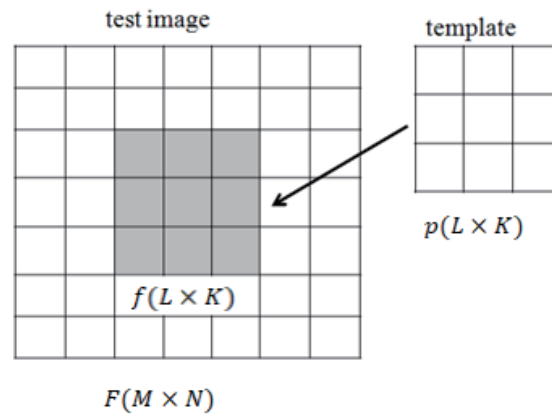


Fig. 6. Match pattern.

to 1. When R is equal to 1, it means that $p(x, y)$ is equal to $f(x, y)$. Examples of sample models and a match pattern are respectively shown in Figs. 7 and 8.

$$R(x, y) = \frac{\sum_0^{L-1} \sum_0^{K-1} (f(x, y) \times p(x, y))}{\sqrt{\sum_0^{L-1} \sum_0^{K-1} (f(x, y))^2} \times \sqrt{\sum_0^{L-1} \sum_0^{K-1} (p(x, y))^2}} \quad (9)$$

The 2D correlation coefficient toolbox, abbreviated as corr2 ,⁽²⁵⁾ is one of the image processing toolboxes in MATLAB. It is mainly used to calculate the correlation coefficient between A and B , where A and B are the matrices or vectors of the same size and can be digital or logical. The return value r is a scalar. The corr2 toolbox computes the correlation coefficient

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right) \left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}}, \quad (10)$$

where \bar{A} is the average of matrix A , \bar{B} is the average of matrix B , m denotes the rows in the matrix, and n denotes the columns in the matrix. Through the match pattern, the number on the smartphone screen can be obtained. Then, we can tell whether the number is correct by sorting row by row, as shown in Fig. 9. If the number is not the preset number, then it is deleted.

4. Control Scheme

This study uses the IBVS combined with fuzzy control to control the robot's position. Through visual servoing, we can instantly update the error with the target position. Then the fuzzy control can effectively reduce the position control error. The visual servo control system

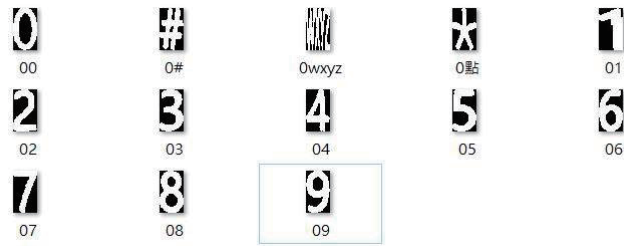


Fig. 7. Sample models.

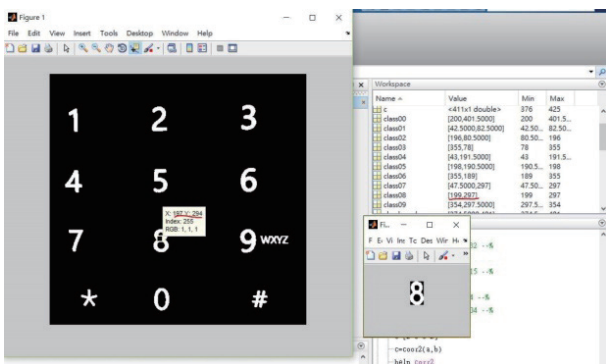


Fig. 8. (Color online) Match pattern of number 8.

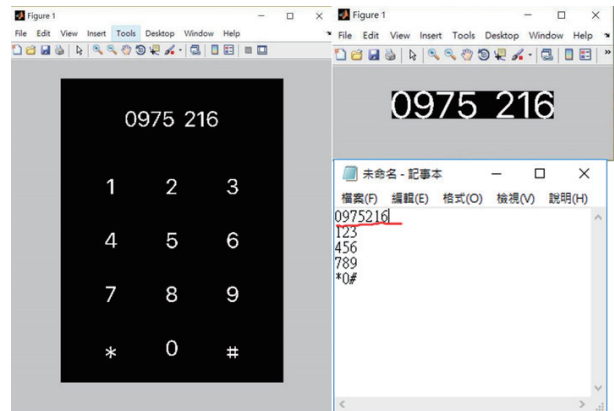


Fig. 9. (Color online) Checking of the number.

architecture is divided into two kinds: position-based and image-based. The position-based architecture uses a vision system to observe the position of the target object in the real space, and compares the target object with the position of the end effector of the robot arm to obtain the position error, that is, the distance between the end effector and the target object in the real space. The position-based visual servo is abbreviated as PBVS. The image-based architecture uses the visual system's target object and captures image feature points to estimate the error between the image feature point location and the robot's arrival location, and then instructs the robotic arm to correct the error. In this study, the IBVS is used as the architecture. A geometric model of the object in the space must be established in advance for the PBVS, and the model is used to estimate the position of the target object. However, if the position of the target object in the geometric model is inaccurate relative to the actual position, the robot arm may not reach the predetermined position. The IBVS is directly controlled in the image feature space and does not require position estimation as shown in Fig. 10.⁽⁸⁾ The camera is on the robot arm.

Fuzzy control is a kind of intelligent computer control based on fuzzy set theory, fuzzy linguistic variables, and fuzzy logic reasoning. Fuzzy control has become a very popular method of performing controller design tasks because it can translate human skills into linguistic rules. The entire process of fuzzy position control is shown in Fig. 11.

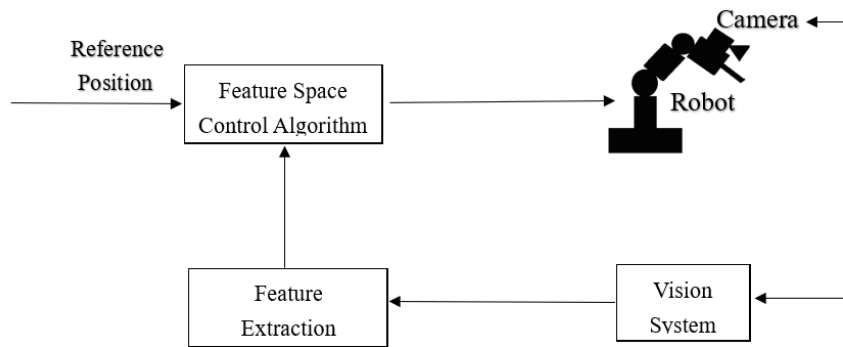


Fig. 10. IBVS visual servo architecture.

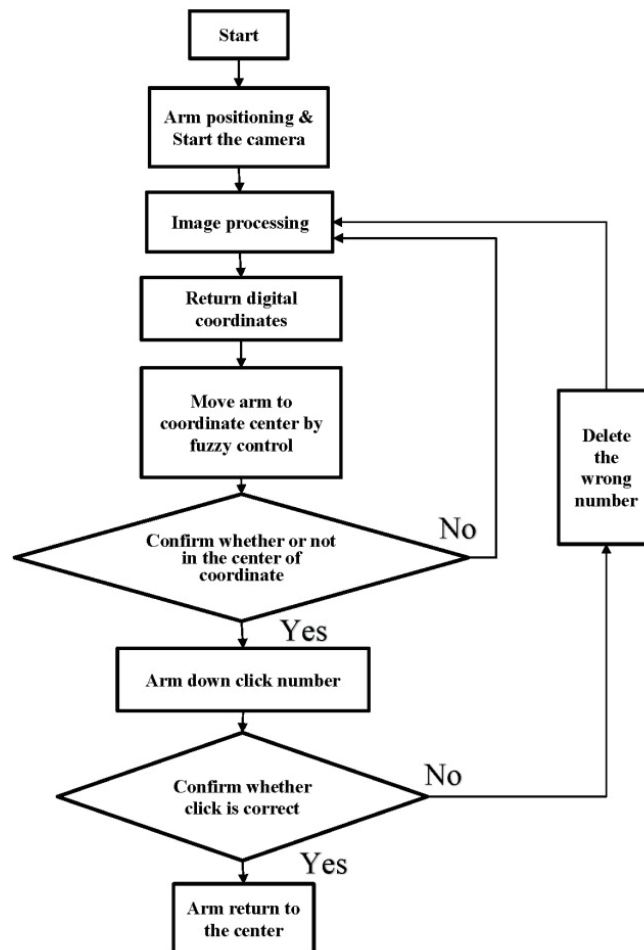


Fig. 11. Flowchart of the control sequence.

Fuzzy theory is used to reduce the error between the commanded and returned degrees of servo motors. It does not require a complex mathematical model of the robot arm. The use of the fuzzy system effectively reduces the number of fuzzy inputs and outputs layer by layer. The fuzzy rules are given in Table 2.

Table 2
Fuzzy rules.

Rule no.	Inputs		Outputs			
	Xe	Ye	Joint-1	Joint-2	Joint-3	Joint-4
1	P	PB	P	PB	PB	PB
2	P	PS	P	PS	PS	PS
3	P	Z	P	Z	Z	Z
4	P	NB	P	NB	NB	NB
5	P	NS	P	NS	NS	NS
6	N	PB	N	PB	PB	PB
7	N	PS	N	PS	PS	PS
8	N	Z	N	Z	Z	Z
9	N	NB	N	NB	NB	NB
10	N	NS	N	NS	NS	NS
11	Z	PB	Z	PB	PB	PB
12	Z	PS	Z	PS	PS	PS
13	Z	Z	Z	Z	Z	Z
14	Z	NB	Z	NB	NB	NB
15	Z	NS	Z	NS	NS	NS

In the table, Xe is the X-axis error, Ye is the Y-axis error, P is positive, N is negative, B is big, and S is small. The fuzzy sets are NB, NS, Z, PS, and PB, which represent negative large, negative small, zero, positive small, and positive big, respectively. The fuzzy control scheme is shown in Fig. 12.

The robot system can calculate the position in real time by automatically updating the coordinate information and the fuzzy controller. The fuzzy system is used to reduce the error between the command data and the returned data. Fuzzy controllers do not require complex mathematical models of robots. Automatically updating coordinate information can reduce interference with the system. Using the D–H model, we can calculate the degrees of the joint motors. However, there is still a slight error when the robot arm reaches the specified position. To overcome this problem, we use the fuzzy controller to adjust the degree of command. Therefore, through these methods, we can drive the robot arm to successfully reach the target button. As shown in Fig. 13, θ_1 , θ_2 , θ_3 , θ_4 , and θ_5 determine the length of r . The coordinates of the robot end effector can be moved to the desired coordinates (d_x, d_y, d_z) by the fuzzy control system.

5. Experimental Results

We perform three tests on the motion of the robot arm, which are *making a call*, *making a call on a different phone*, and *deleting a wrong number*. Each experiment will be explained under each subtitle. Figure 14 shows the two smartphones that are tested in this experiment. They are an HTC smartphone and a Zenfone3. Table 3 shows the specifications of the two smartphones.

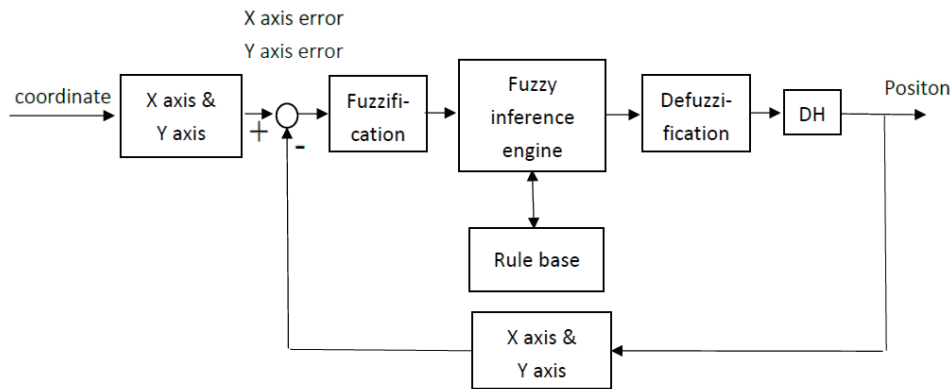


Fig. 12. Fuzzy control scheme.

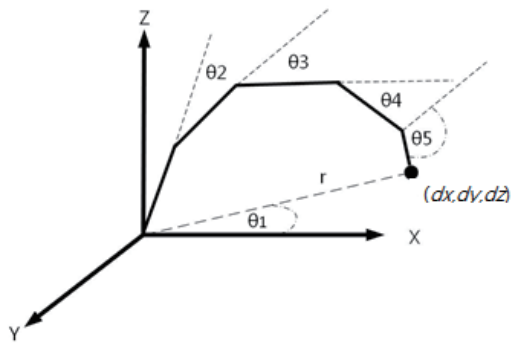


Fig. 13. Joint relationship of the robot arm.



Fig. 14. (Color online) Two smartphones tested in this experiment.

Table 3
Specifications of the smartphones.

Type of smartphone	HTC	Zenfone3
Screen size (inch)	4.7	5.2
Type of touch panel	Capacitive touch	Capacitive touch
Length (mm) × width (mm) × thickness (mm)	131.5 × 70.7 × 9.9	106 × 55.2 × 14.6
Character size in X and Y directions (cm)	1.4 × 2	1.7 × 2.1

A. Making a call

This experiment demonstrates the use of machine vision to guide the robot arm to click on the desired numbers 0975216653, as shown in Fig. 15. First, the signal is transmitted through the notebook, then the robot arm receives the signal and starts performing the specified motion. After the robot arm completes the entire procedure and confirms the numbers are correct, the arm makes a telephone call.

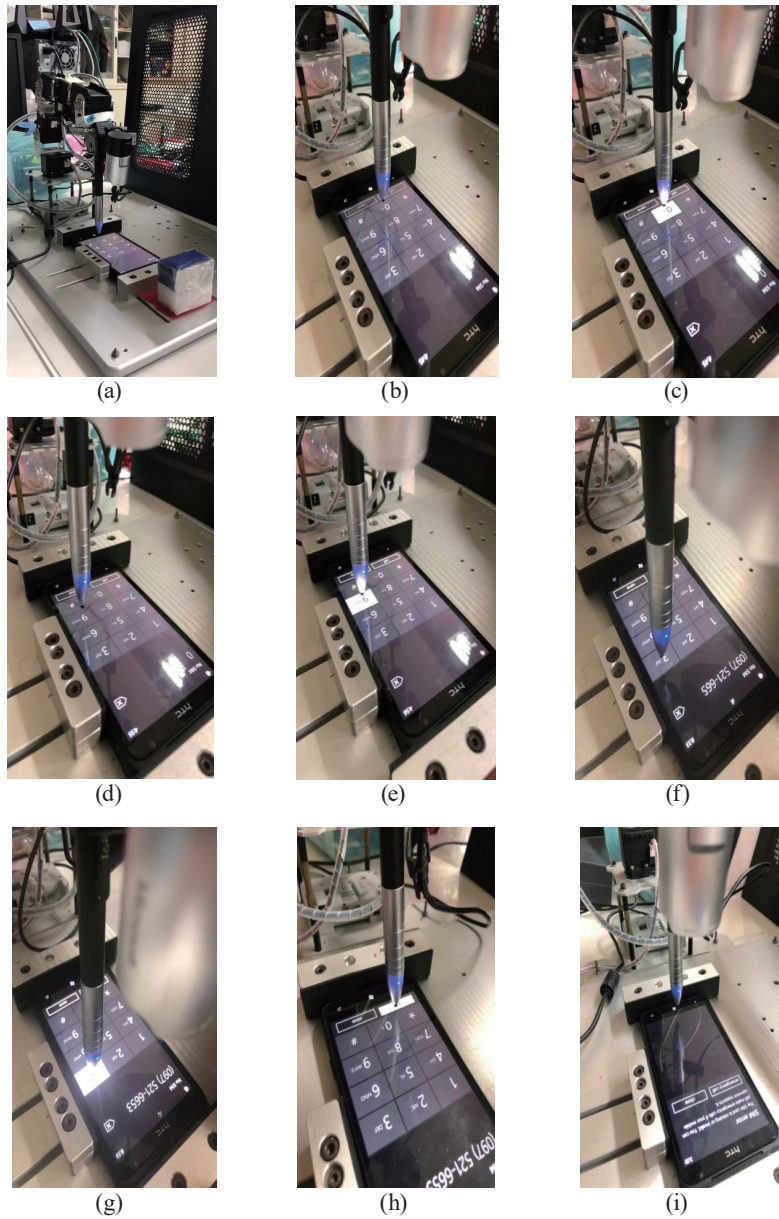


Fig. 15. (Color online) Making a call. (a) Starting point. (b) Find the number 0. (c) Click on the number 0. (d) Find the number 9. (e) Click on the number 9. (f) Find the number 3. (g) Click on the number 3. (h) Click on the call button. (i) Complete dialing.

B. Making a call on a different phone

The next experiment demonstrates the use of another smartphone and uses machine vision to click on the numbers 0975216653, as shown in Fig. 16. First, the signal is transmitted through the notebook, then the robot arm receives the signal and starts performing the specified motion. After the robot arm completes the entire procedure and confirms the numbers are correct, the arm makes a telephone call.

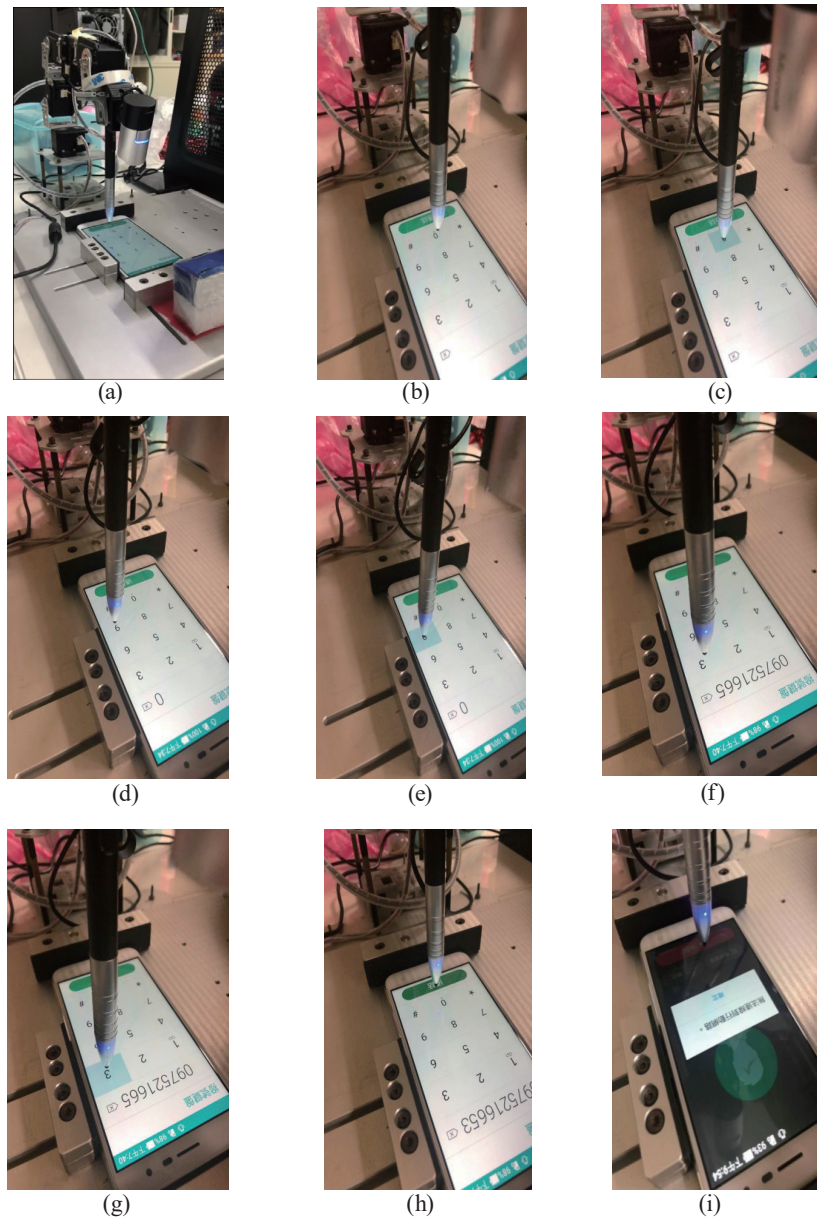


Fig. 16. (Color online) Making a call on a different phone. (a) Starting point. (b) Find the number 0. (c) Click on the number 0. (d) Find the number 9. (e) Click on the number 9. (f) Find the number 3. (g) Click on the number 3. (h) Click on the call button. (i) Complete dialing.

C. Deleting a wrong number

The third experiment demonstrates that the visual servo robot control system can delete a wrong number, as shown in Fig. 17. We use geometric matching to determine whether the number is correct in every step. Here, the number 7 is pressed twice, and the robot arm can delete the extra number 7.

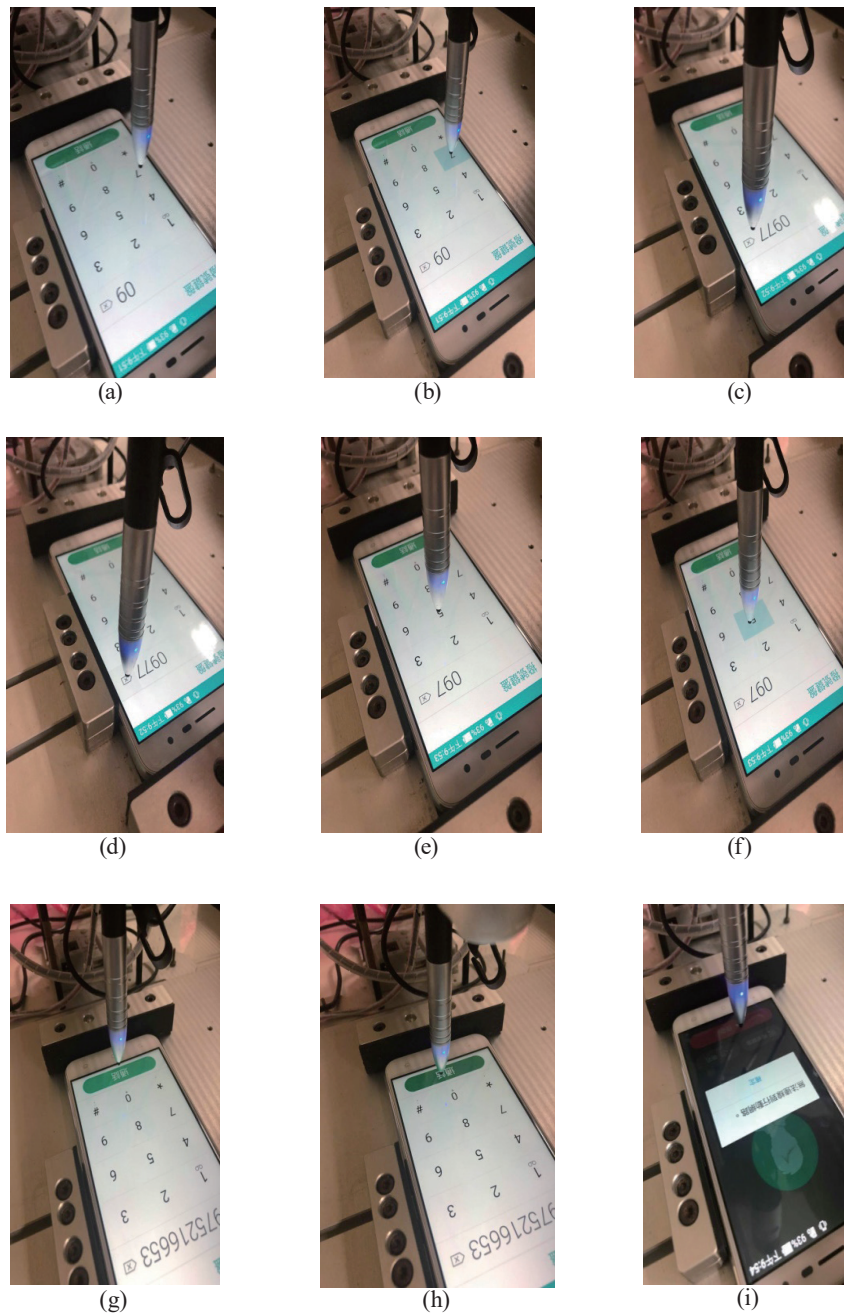


Fig. 17. (Color online) Deleting a wrong number. (a) Find the number 7. (b) Click on the number 7 twice. (c) Find the delete key (d) Remove the redundant number 7. (e) Find the number 5. (f) Click on the number 5. (g) Find the call button. (h) Click on the call button. (i) Complete dialing.

Figure 18 shows the result of clicking the number 3. The positioning fuzzy control of the Y -axis and X -axis is good, and the position error converges to zero during the fuzzy control process.

In a previous work,⁽⁷⁾ similar tests were performed using a fixed camera; thus, mistakes had to be canceled and repointed. The present study applies eye-in-hand visual servoing control in a

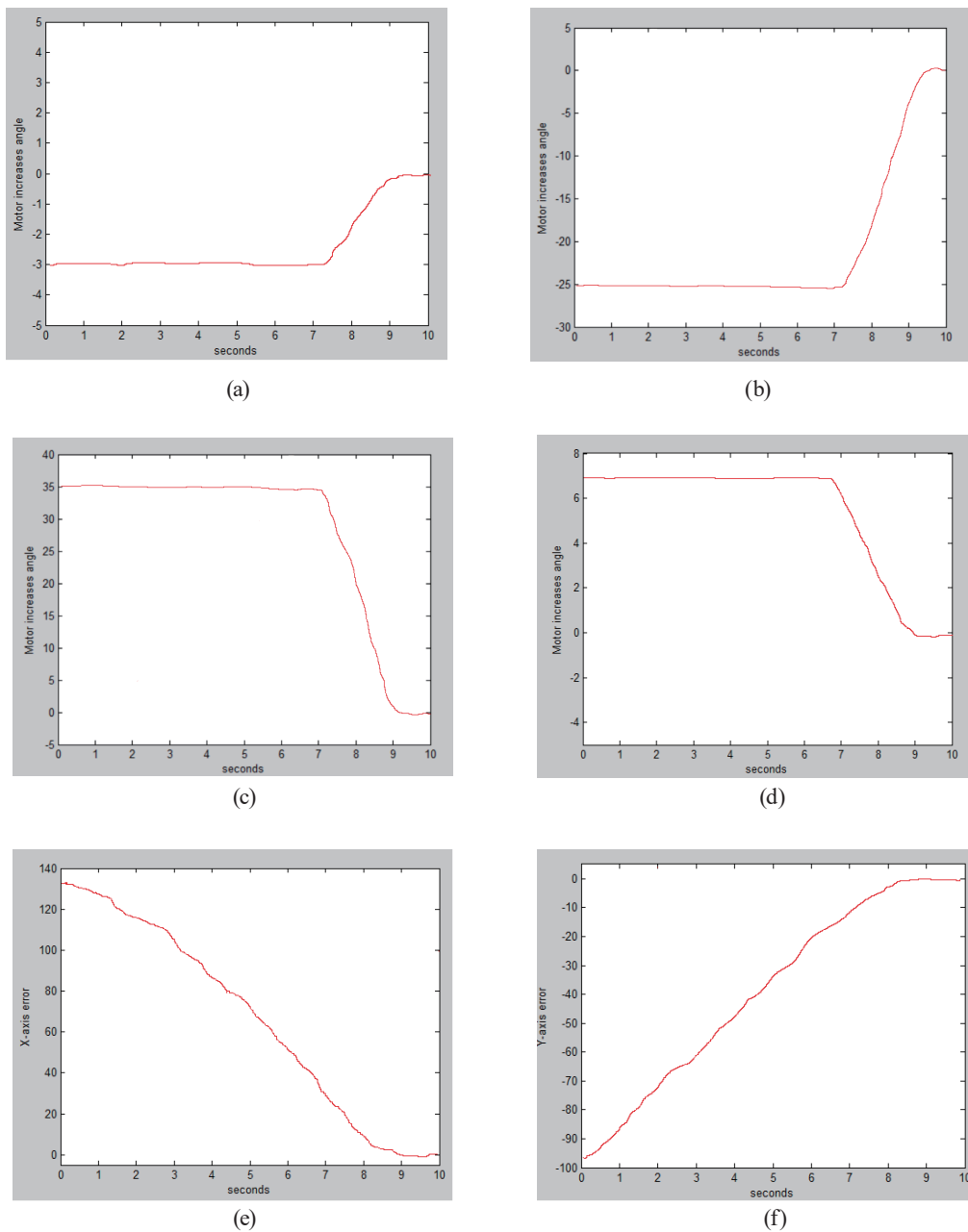


Fig. 18. (Color online) Fuzzy control signals and position errors when clicking the number 3. (a) Fuzzy control signal of Joint-1. (b) Fuzzy control signal of Joint-2 (c) Fuzzy control signal of Joint-3. (d) Fuzzy control signal of Joint-4. (e) X -axis error. (f) Y -axis error.

closed-loop system. Thus, there are no such mistakes that need to be canceled and reprinted. The error of an edge click can be corrected through the visual servo in real time through the closed-loop control system. The proposed fuzzy rules are performed in the closed-loop control system, and the convergence of the output error is guaranteed by the IBVS.

6. Conclusions

We propose an intelligent control scheme based on image processing, pattern recognition, character recognition, and a fuzzy system to control the position of a robot arm. Inverse kinematics, conversion between webcam coordinates and robot arm coordinates, and fuzzy rules are used to capture numbers or letters outside the target center. The inverse kinematics solution is obtained by the D-H model. In image processing, the image in the RGB color space is converted into the HSV color space and then binarized. The noise in the binary image is excluded. Finally, an optical character recognition program is implemented through pattern recognition, where characters are corrected using a prestored database. The character recognition process is performed using features in the training data. The program can distinguish between the 10 Arabic numerals. In the control scheme, the proposed system can obtain the position in real time using a visual servo, the D–H model, and a fuzzy controller. By converting the coordinates at each interval through the D–H model, the degrees of the joint are updated by visual servoing. The experimental results show that the proposed control scheme can make the robot arm successfully execute different smartphone testing mechanisms. Most importantly, the robot arm can handle situations when the smartphone is being moved slightly, and camera calibration is not needed.

References

- 1 K. Zheng, P. Cui, and H. Mao: Proc. 29th Chinese Control Conf. (2010).
- 2 Z. Chi and Q. Xu: Proc. 10th IEEE Int. Conf. Nana/Micro Engineered and Molecular Systems (2015).
- 3 Y. H. Lu and J. G. Juang: *Sens. Mater.* **30** (2018) 1643.
- 4 Y. H. Weng: Automatic Robot Assembly with Eye-in-Hand Binocular Visual Servoing and Structured Lighting, Master Thesis, Department of Electrical Engineering, National Taipei University of Technology (2011).
- 5 X. Gao, C. Zhou, F. Chao, L. Yang, C. Lin, and C. Shang: *IEEE Access* **7** (2019) 144043.
- 6 R. Wu, C. Zhou, F. Chao, L. Yang, C.M. Lin, and C. Shang: *Neurocomputing* **388** (2020) 12.
- 7 J. G. Juang, Y. J. Tsai, and Y. W. Fan: *Appl. Sci.* **5** (2015) 851.
- 8 G. Q. Zhang: Master Thesis Application of Optical-Flow Algorithm in Visual-Servoing Manipulators (National Taiwan Ocean University, 2012).
- 9 X. Cao, Q. Zhang, and Z. Ni: Proc. 2nd Int. Conf. Robotics and Automation Engineering (2017).
- 10 X. Xiao, Y. Li, H. Tang: Proc. IEEE Int. Conf. Robotics and Biomimetics (2017).
- 11 Y. Zhang, Y. Yu, S. Jia, and X. Wang: Proc. 36th Chinese Control Conf. (2017).
- 12 G. S. Huang, C. K. Tung, H. C. Lin, and S. H. Hsiao: Proc. IEEE Int. Conf. Robotics & Automation (2011).
- 13 Y. S. Peter, C. Muller, and C. Zheng: *IEEE Trans. Control Syst. Technol.* **18** (2010) 1280.
- 14 H. Wang, Y. H. Liu, and W. Chen: *IEEE Trans. Control Syst. Technol.* **18** (2010) 1359.
- 15 Y. Chen, L. Liu, M. Zhang, and H. Rong: Proc. 6th World Congr. Intelligent Control and Automation (2006).
- 16 C. R. Rocha, C. P. Tonetto, and A. Dias: *Rob. Comput.-Integr. Manuf.* **27** (2011) 723.
- 17 J. J. Craig: *Introduction to Robotic: Mechanics & Control* (Addison-Wesley, Boston, 1989).
- 18 L. W. Tsai: *Robot Analysis – The Mechanics of Serial and Parallel Manipulator* (Wiley InterScience, Hoboken, New Jersey, 2005).
- 19 S. B. Niku: *Introduction to Robot-Analysis, System, Application* (Prentice Hall, Hoboken, New Jersey, 2004).
- 20 S. Hutchinson, G. D. Hager, and P. I. Croke: *IEEE Trans. Rob. Autom.* **12** (1996) 651.
- 21 Y. C. Liao: Master Thesis, Visual Servo Control and Applications of Robotic Manipulators Based on MATLAB (National Taiwan University of Science and Technology, 2010).
- 22 E. Dubois: *The Structure and Properties of Color Spaces and the Representation of Color Images* (Morgan & Claypool, California, USA, 2009).
- 23 D. Liu, H. Mansou, P. T. Boufounos, and U. S. Kamilov: Proc. IEEE Int. Geoscience and Remote Sensing Symp. (2018).
- 24 I. H. Cheng: Master Thesis, Application of 3D Coordinate and Real-time Character Recognition for 5DoF Robotic Arm on Smartphone Automatic Test System (Nation Taiwan Ocean University, 2015).
- 25 Corr2: <https://www.mathworks.com/help/images/ref/corr2.html> (accessed December 2019).