**Indonesian Journal of Electrical Engineering and Computer Science**
Vol. 6, No. 1, April 2017, pp. 72 ~ 79
DOI: 10.11591/ijeecs.v6.i1.pp72-79


# Intelligent Control for Visual Servoing System

**Dwi Pebrianti*[1], Ong Ying Peh[2], Rosdiyana Samad[3], Mahfuzah Mustafa[4], N.R.H Abdullah[5], Luhur Bayuaji[6]**
[1,2,3,4,5]Faculty of Electrical and Electronics Engineering
[6]Faculty of Computer Sience and Software Engineering
Universiti Malaysia Pahang, Pekan Campus, 26600 Pekan, Pahang, Malaysia
Phone: +60-9-424-6127; Fax: +60-9-424-6111
*Corresponding author, e-mail: dwipebrianti@ump.edu.my

### *Abstract*

*This paper presents intelligent control for visual servoing system. The proposed system consists of a camera placed on a Pan Tilt Unit (PTU) which consists of two different servo motors. Camera and PTU are connected to a personal computer for the image processing and controlling purpose. Color threshold method is used for object tracking and recognition. Two different control methods, PID and Fuzzy Logic Control (FLC) are designed and the performances are compared through simulation. From the simulation result, the settling time of PID controller is 40 times faster than FLC. Additionally, the rise time of PID is about 20 times faster than FLC. However, the overshoot percentage of PID controller is 4 times higher than FLC. High overshoot value is not preferable in a control system, since it will cause the damage to the system. Real implementation of FLC on a home-built visual servoing system is conducted. Two different types of FLC, 9 and 11 rules of FLC are designed and implemented on the system. The experimental result shows that FLC with different total number of rules give different system performance. The settling time of FLC with 11 rules is 2 times faster than FLC with 9 rules. Additionally, the overshoot percentage of FLC with 11 rules is 2 times lower than FLC with 9 rules.*

***Keywords***: *PID controller, fuzzy logic controller, visual servoing system, image processing, object tracking*



## 1. Introduction

Recently, intelligent visual servoing system had assimilated into human daily life unconsciously such as automotive cruise control, robotic arm, production lines and other industrial applications. Object tracking and grasping based on intelligent control visual sensors are the most recent research topics [1] as the system is able to work under any unpredictable and hazard circumstances such as in chemical reactive area. Robot is preferable to be used instead of manpower because robot is able to decrease the risk for human life in performing dangerous tasks [2].

Visual servoing system or sometimes called as visual feedback system is a system that fuses many elements, such as image processing, kinematics, dynamics, control theory and real-time computing. Visual servoing also provides the potential to relax the mechanical accuracy and stiffness requirements for robot mechanisms and hence reduce their cost. Another advantage to incorporate vision in a control system is to increase the flexibility of a control system. Additionally, the main aim of the image based visual servoing is to control the end-effector of a robot arm such as a set of image features to reach a desired configuration [3].

Controlling a visual servoing system is a complex task. Burlacu [4] developed Model Predictive Controller (MPC) for a visual servoing system.The system is divided into two loops, inner and outer loop. The plant used is a 6 d.o.f robot. The challenge in this work is the difficulty to generate the control law. A set of mathematical equation that relates to the system dynamics must be obtained first, before constructing the control law. Hsu et al. [5] designed an adaptive controller to allow trajectory tracking on smooth surfaces non-orthogonal to the optical axis. Since the system model for visual servoing is nonlinear due to the depth displacement, an approximate linearly parameterized function representing the system was used in order to design a suitable linearly parameterized adaptive control law. The dependence of depth with respect to the 2D image coordinates was then adaptively compensated without measuring the depth.

*Received November 2, 2016; Revised January 25, 2017; Accepted February 15, 2017*

Intelligent control had been widely used in many applications, for example in medical field, automation, industrial process, etc. The ability to mimick human behavior in intelligent control makes this control scheme being so popular. The application of intelligent control in visual servoing system has been widely used in many industrial processes. The intelligent control in visual servoing system had been proposed in [6]. The Fuzzy Logic Controller (FLC) had been proposed by Kim and Seo [7]. The inputs and output for the FLC were based on the pixel values. While the rule base were set based on human knowledge.

This paper proposes a method for controlling a visual servoing system base on intelligent control. The first step is to find a mathematical model of a home-built visual servoing system by using System Identification in MATLAB. Two different methods, Proportional Integral Differential (PID) controller as a conventional method and Fuzzy Logic Control (FLC) as an intelligent control will be designed and the performance will be compared by using simulation in Simulink, MATLAB. Lastly, real-time implementation of the designed intelligent controllers will be conducted.

The remainder of this paper is structured as follows. Section 2 explains the experimental setup used in this research. Section 3 describes the research method which covers the image processing, mathematical model of the Pan Tilt Unit (PTU) and the design of FLC. Section 4 will be the result and discussion and Section 5 will be the conclusion and future works.

## 2. Experimental Setup

A home-built visual servoing system consists of a web camera, a Pan Tilt Unit (PTU) and an Arduino microcontroller. The pan-tilt servo kit is resembled and the servo motors of the PTU are connected to the Arduino Uno board. The camera is attached on the servo motor for the object detection. Camera and microcontroller are connected to a personal computer for the image processing and controlling purpose. Figure 1 shows the home-built hardware prototype. The C++ programming is written in the Microsoft Visual Studio 2013 for the FLC. The best tuned FLC in simulation will be implemented on hardware.

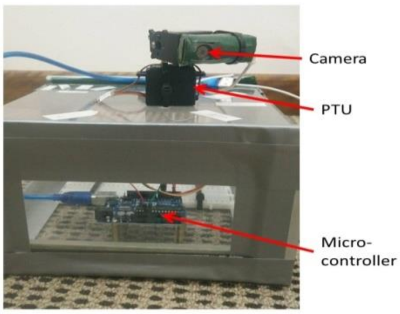


Figure 1. Home-built Visual Servoing System

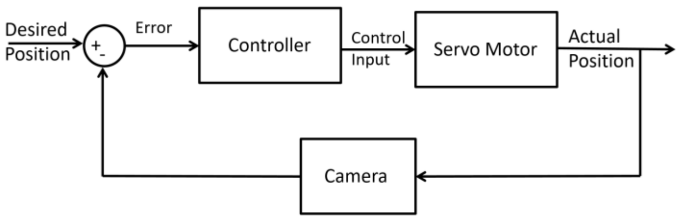


Figure 2. Block Diagram of The System

## 3. Research Method

### 3.1. Object Tracking

The object tracking is based on color detection. The image captured by using a web camera is in RGB color format. This RGB image is converted into HSV color image. In order to extract the red target from other objects, color threshold is applied. This threshold is adjusted by trial and error depending on the level of illumination during the experiment. Once the image is thresholded, then it will be filtered by using Gaussian smoothing method. This filtering purpose is to remove or reduce any noises appear in the image so that it will give a clearer image of the object. The image result after filtering process is in binary image format. This binary image is used for the Center of Gravity (COG) calculation. The COG represents the position of the target. The calculation of the COG of an object is based on the common momentum.The value of CoG obtained will be used by the controller to adjust the position of servo motor so that the camera will continuously follow the target.

### 3.2. Mathematical Model of Servo Motor

DF05BB Pan Tilt servo motor is used for moving the mounted camera. The servo motor for pan movement is identical with the tilt movement. Therefore, the strategy used here is to find the mathematical model of one servo motor and use this model to design the controller for both pan and tilt movement. The rotation range for this motor is from 0° to 180°. The motor signal wire is connected to Arduino Uno board in order to control the movement. The tilt movement is controlled by the upper servo motor while the pan movement is controlled by the lower servo motor.

The transfer function of the DF05BB servo motor is identified by using System Identification Toolbox in Matlab. The input for the servo motors is PWM signal and the output is the rotational movement of the motor. The input and output data is obtained by observing the output graph from an oscilloscope which is connected to the hardware. The PWM signals shown in the oscilloscope and the real angles of servo motor movement are recorded. These values are then used as the input and output values in System Identification Toolbox. It is found that the best mathematical model achieved is a third order system, which is expressed in Equation (1).

$$G(s) = \frac{96.83}{(10.04s+1)(0.00085s+1)(0.00037s+1)} \tag{1}$$

### 3.3. Controller Design

Two different controllers are designed in this study. The first one is PID controller and the second one is Fuzzy Logic Controller (FLC). The block diagram of the system is shown in Figure 2. The controller will be represented by PID controller or FLC.

#### 3.3.1. PID Controller

The control input for PID controller is shown in Equation (2).

$$U(s) = K_p + K_d s + \frac{K_i}{s} \tag{2}$$

where $K_p$, $K_d$ and $K_i$ are values of proportional gain, differential gain and integral gain, respectively.

Initially, the value for $K_p$ is set into 1 while the value of the $K_d$ and $K_i$ are set into 0 for closed-loop response. Subsequently, the gains are tuned individually in order to get the optimum performance. The gains are tuned by increasing the $K_p$ by 10 for each time tuning while for the $K_i$ value is increased by 10 for each tuning process. Lastly, the $K_d$ is tuned by increasing 0.01 for each time tuning process. The gains are tuned until achieving the optimum performance where the overshoot, settling time and rise time are minimized.

#### 3.3.2. Design of Fuzzy Logic Controller

Fuzzy Logic Controller (FLC) consists of three steps, which are fuzzification, knowledge base rule design and defuzzification process.

### 3.4. Fuzzification

In this step, the linguistic variables are converted to fuzzy variables. For position control system, there are three fuzzy variables which are position error, $e$, delta-error, $\Delta e$ and Pulse Width Modulated (PWM). The position error and the delta-error are the inputs for the FLC while the PWM is the output. In this study, 5 membership functions for the position error, $e$ input as shown in Figure 3 are constructed. 3 membership functions for delta-error input $\Delta e$ as shown in Figure 4 and PWM output with 5 membership functions as shown in Figure 5 are designed.

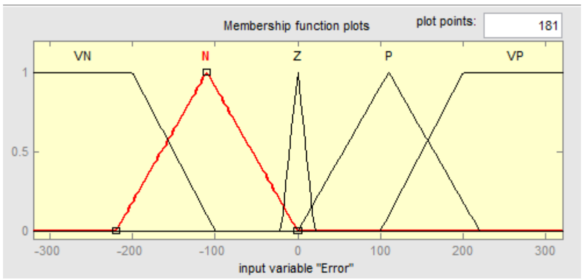


Figure 3. Membership function for position error input

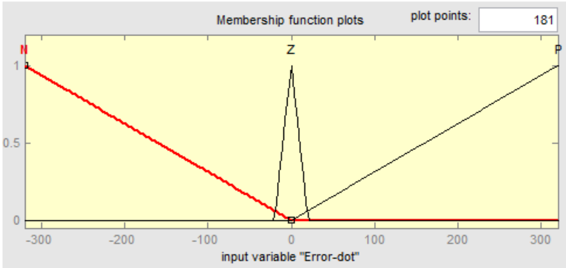


Figure 4. Membership function for delta-error input

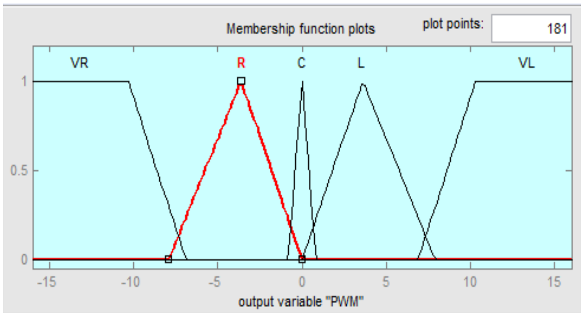


Figure 5. Membership function for PWM output

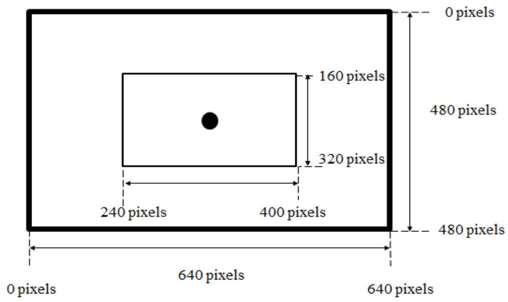


Figure 6. Illustration for the frame center

### 3.5. Knowledge Base

11 rules are designed for the system. The rules are set according to the movement and position of the target object. By referring to Figure 6, the system needs to maintain the object to be at the center of an image. Thus, the negative error input means the target object is at the right side of the camera view. In this case, the visual servo needs to move to the right. Additionally, the positive delta-error means that the target object is moving to right side and this will cause the camera to move to the right in order to maintain the target is in the center of the image. This knowledge base is then translated into a set of rules as listed in Table 1.

Table 1. Fuzzy Rules Used in The Design

| Rule No. | Description |
|---|---|
| 1 | IF error is VN THEN PWM is VR |
| 2 | IF error is N THEN PWM is R |
| 3 | IF error is P THEN PWM is L |
| 4 | IF error is VP THEN PWM is VL |
| 5 | IF error is Z and delta-error is N THEN PWM is L |
| 6 | IF error is Z and delta-error is P THEN PWM is R |
| 7 | IF error is VN or delta-error is P THEN PWM is R |
| 8 | IF error is VP or delta-error is N THEN PWM is L |
| 9 | IF error is Z and delta-error is Z THEN PWM is C |
| 10 | IF error is N or delta-error is P THEN PWM is R |
| 11 | IF error is P or delta-error is N THEN PWM is L |

### 3.6. Defuzzification

In this step, a crisp output value will be generated for the position control mechanism. The CoG method is applied for this project as the Fuzzy Inteference System (FIS) used is Mamdani FIS. The CoG method can be expressed as:

$$Y = \frac{\sum_{j=1}^{n} f_A(w_j)\cdot w_j}{\sum_{j=1}^{n} f_A(w_j)} \quad (3)$$

## 4. Result And Analysis

### 4.1. Object Tracking Result

Figure 7 shows the result of RGB image capture, HSV image and object detection by using color as the feature. Figure 8 shows the result of COG calculation. It is clearly seen that the result of object tracking is performing well.

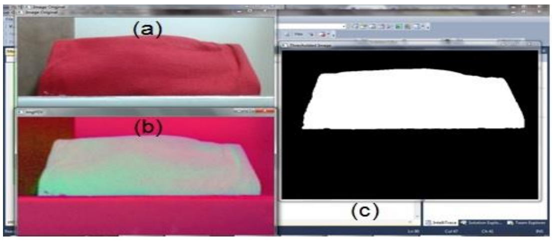


Figure 7. (a) RGB Image (b) HSV Image (c) Thresholded Image

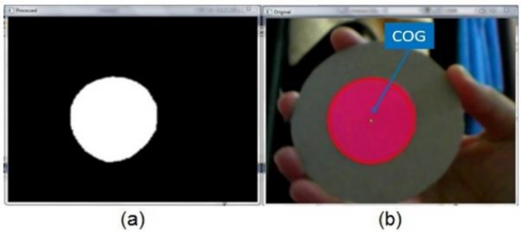


Figure 8. (a) Binari Image (b) Center of Gravity of the Target

### 4.2. Performance Comparison Between PID Controller And 11 Rules FLC

The comparison between PID controller and Fuzzy Logic Controller (FLC) is done by using Simulink in MATLAB. Figure 9 shows the comparison between PID controller and 11 rules FLC. After several simulation for tuning process, the best gain for PID controller is $K_p = 200, K_i = 20$ and $K_d = 0.25$. Additionally, it is clearly seen that the overall performance for 11 rules FLC is better than PID controller. FLC has lower overshoot percentage which enables the controller to perform well. High overshoot percentage will damage the servo motor. The rise time for PID is shorter than FLC which means that PID is faster than FLC. However, in control problem, faster response is not the only answer. The appropriate combination between fast response time and low overshoot will be preferable. In this case, FLC with 11 rules is considered to be better than PID controller.

### 4.3. Comparison Between 11 Rules and 9 Rules FLC

In FLC with 9 rules, the 11 rules FIS is modified. Rule 1 and Rule 4 are removed from the controller while the shapes for the membership function remain unchanged. Figure 10 shows the comparison between FLC with 9 and 11 rules. From this figure, it is seen that the FLC with 9 rules is never reaching steady state as it remains fluctuating over time. The transient response parameters shown in Figure 10 are for the FLC with 9 rules. The time response for FLC with 11 rules is mentioned in Figure 9. According to both graphs, FLC with 9 rules is obviously having higher overshoot percentage compared with the 11 rules FLC.

From the comparisons, it can be concluded that the 11 rules FLC has the best performance. This is because FLC with 11 rules has reached steady state condition. It is also observed that the more rules used in FLC, the more accurate and precise the result is. However, it should be noted that the more rules will give effect to the processing time. Therefore, the total number of rules used in FLC must be designed properly.

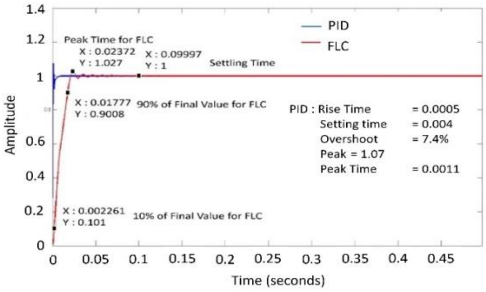


Figure 9. Comparison between PID controller and 11 rules FLC

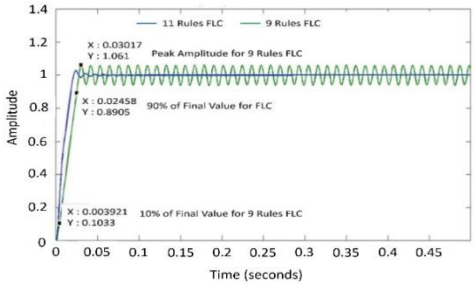


Figure 10. Comparison of Performance for 11 rules FLC and 9 rules FLC

### 4.4. Hardware Implementation

Based on the simulation result, FLC is implemented on home-built visual servoing system. The experiment is conducted to check the performance of the designed FLC. The analysis is mainly conducted whether the servo system can follow the movement of an object or not and the time response performance in term of overshoot percentage and settling time.

At data acquisition stage, the target object was moved according to a sequencial direction, **right** → **left** → **up** → **down**. The target object was hold in a position for a short period of time (1 minute) in order to observe the stability of the system. Furthermore, the graphs were plotted according to the x (horizontal) and y (vertical) axis of the servo movements. Figure 11 and Figure 12 show the graphs for the real time performance of the 11 rules FLC in x-axis movement and y-axis movement, respectively.

Additionally, for the comparison purpose, 9 rules FLC is also implemented on hardware. The performance is analyzed and compared with 11 rules FLC. Figure 13 and Figure 14 show the graphs for the real time performance of the 9 rules FLC in x-axis movement and y-axis movement, respectively.

Table 2 shows the time response analysis for both FLC for x (horizontal) movement and y (vertical) movement. It is seen that FLC with 11 rules has lower overshoot percentage and faster setting time compared to FLC with 9 rules. This experimental result is agree with the simulation result which shows that 11 rules FLC has better performance compared to 9 rules FLC.

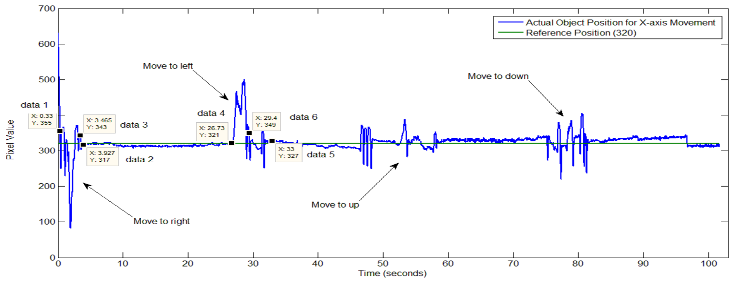


Figure 11. Real Time Performance of the 11 Rules FLC in x-axis Movement

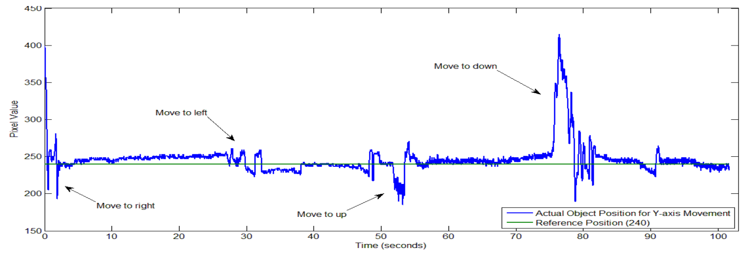


Figure 12. Real Time Performance of the 11 Rules FLC in y-axis Movement

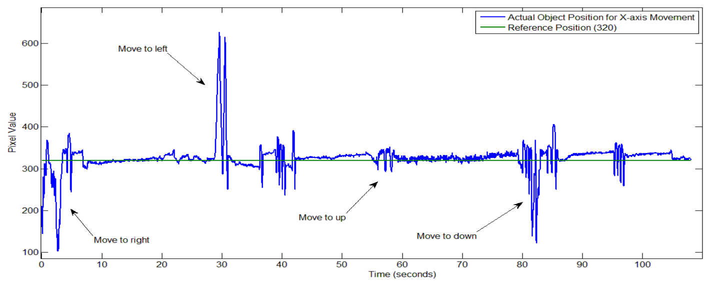


Figure 13. Real Time Performance of the 9 Rules FLC in x-axis Movement

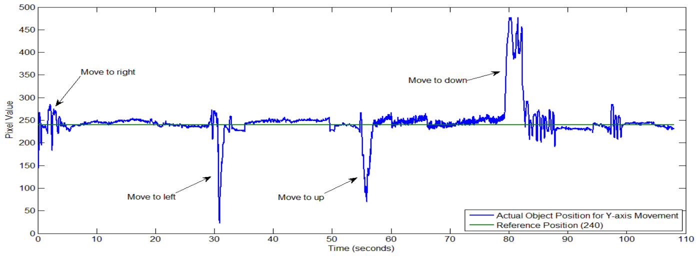


Figure 14. Real Time Performance of the 9 Rules FLC in y-axis Movement

Table 2. 9 and 11 Rules FLC Comparison

| Number of Rules | X-axis movement | | | | Y-axis Movement | | | |
|---|---|---|---|---|---|---|---|---|
| | To Right | | To Left | | To Up | | To Down | |
| | Settling time (sec) | Over-shoot (%) | Settling time (sec) | Over-shoot (%) | Settling time (sec) | Over-shoot (%) | Settling time (sec) | Over-shoot (%) |
| 11 Rules | 3.60 | 7.19 | 6.27 | 9.06 | 6.80 | 4.58 | 9.50 | 5.42 |
| 9 Rules | 7.46 | 7.81 | 15.28 | 11.56 | 10.66 | 2.08 | 11.25 | 13.75 |

## 5. Conclusion And Future Works

From the simulation for PID and Fuzzy Logic Controller (FLC), the system performance for FLC is better than PID controller. This is due to the intelligent rules set in the FLC enable the system to perform in more precise way. The rule base and parameters of membership function plays important roles in fuzzy controller. By making a good choice of rule base and parameters of membership function, the FLC system can perform better than PID controller. The input membership function in fuzzy logic enables the FLC to handle imprecise inputs while PID controller must need a precise input for achieving a certain reference.

The 11 rules FIS has the best performance compared to other parameters. By deleting the rules in knowledge base, it will cause the tracking process becomes imprecise. This is due to some inputs are not evaluated during the tracking process.

Experimental result was conducted to prove the performance of the designed FLC on a home-built visual servoing system. The result shows that 11 rules FLC has the better performance compared to 9 rules FLC. As for the future work, other shapes of membership function for the input and output of FLC will be analyzed. Additionally, fully embedded visual servoing system will be developed.

## Acknowledgements

This research is partially supported by Ministry of Higher Education Malaysia under the Exploratory Research Grant Scheme, RDU 130604 and Fundamental Research Grant Scheme, RDU 140137.

## References

[1] I Markovic, F Chaumette, I Petrovic. *Moving object detection, tracking and following using an omnidirectional camera on a mobile robot*. 2014 IEEE International Conference Robot and Automation. Hongkong, China. 2014: 5630–5635.

[2] MA Akbar. *Simulation of Fuzzy Logic Control for DC Servo Motor Using Arduino Based on MATLAB/Simulink.* 2014 International Conference of Intelligent Autonomous Agents, Networks and System. Bandung, Indonesia. 2014: 42–46.

[3] Chaumette F, Hutchinson S. Visual Servo Control, Part II: Advanced Approaches. *IEEE Robotics and Automation Magazine.* 2007; *14*: 109-11.

[4] Adrian Burlacu, Corneliu Lazar. Image Based Controller for Visual Servoing System. *Automatica*. 2008; 58 (1).

[5] Hsu L, Zachi ARL, Lizarralde F. *Adaptive Visual Tracking for Motions on Smooth Surfaces*. IEEE Conference on Decision and Control. Orlando, Florida. 2001: 1866–1871.
[6] YM Zhao, WF Xie, S Member, TT Wang. *Neural Network-Based Image Moments for Visual Servoing of Planar Objects.* Proceeding of IEEE/ASME International Conference on Advance Intelligent Mechatronics. 2012: 268–273.
[7] CS Kim, WH Seol, SH Han, O Khatib. *Fuzzy Logic Control of a Robot Manipulator Based on Visual Servoing*. 2001: 1597–1602.