

# Behavioristic image-based pose control of mobile manipulators using an uncalibrated eye-in-hand vision system

Tsing-Iuan James Tsay<sup>1</sup> · Pei-Jiun Hung<sup>1</sup>

Received: 24 April 2017 / Accepted: 21 August 2017 / Published online: 26 October 2017  
© ISAROB 2017

**Abstract** In the execution of material handling, the mobile manipulator is controlled to reach a station by its mobile base. This study adopts an uncalibrated eye-in-hand vision system to provide visual information for the manipulator to pick up a workpiece on the station. A novel vision-guided control strategy with a behavior-based look-and-move structure is proposed. This strategy is based on six image features, predefined by image moment method. In the designed neural-fuzzy controllers with varying learning rate, each image feature error is taken to generate intuitively one DOF motion command relative to the camera coordinate frame using fuzzy rules, which define a particular visual behavior. These behaviors are then fused to produce a final command action to perform grasping tasks using the proposed behavior fusion scheme. Finally, the proposed control strategy is experimentally applied to control the end-effector to approach and grasp a workpiece in various locations on a station.

**Keywords** Behavior-based · Behavior fusion · Image moment · Neural-fuzzy controller · Vision-guided

## 1 Introduction

Mobile manipulators for material handling are extensively used to transfer material in production lines with short life

cycle, small-volume and wide-variety of products [1]. The guidance control system of the mobile base of the mobile manipulator inevitably causes position and orientation errors of the mobile base. Uncertainties in either the location of the mobile base or the location of the object are responsible for a position mismatch, which normally causes failure in the following pick-and-place operation. A more direct means of solving this problem is to use a CCD camera to provide visual feedback. The camera is generally attached to the end-effector of the manipulator.

The visual servo control structures of such an eye-in-hand manipulator can be classified into four major categories [2] according to the coordinates where the error signal is defined and whether the control structure is hierarchical. The control objective of position-based control is to eliminate errors defined by the pose of the target with respect to the camera, and the features extracted from the image planes are used to estimate the spatial pose of the target. Then, the control law sends the command to the joint-level controllers to drive the servomechanism. This control architecture is termed hierarchical control and is considered a dynamic position-based look-and-move control structure. If the servomechanism is directly controlled by the control law mentioned above instead of the joint-level controller, the control architecture is referred to as a position-based visual servo control structure. Meanwhile, in image-based control, the errors to the control law are determined directly from the extracted features of the image planes. However, if the errors are then sent to the joint-level controller to drive the servomechanism, the control architecture is hierarchical and is called a dynamic image-based look-and-move control structure. On the contrary, the visual servo controller eliminates the joint-level controller and directly controls the servomechanism. This control architecture is termed an image-based visual servo control structure.

---

This work was presented in part at the 22nd International Symposium on Artificial Life and Robotics, Beppu, Oita, January 19–21, 2017.

---

✉ Tsing-Iuan James Tsay  
tjtsay@mail.ncku.edu.tw

<sup>1</sup> National Cheng Kung University, Tainan, Taiwan

Apart from the traditional visual servo control methods mentioned above, behavior-based approaches to visual servo control have already been presented in the literature [3, 4]. An uncalibrated eye-in-hand vision system is adopted to provide visual information to the mobile manipulator to pick up a workpiece located on the station. This paper focuses on the development of the behavioristic image-based visual servoing algorithm for the robot manipulator mounted on the mobile base.

## 2 Image processing

In this study, the workpiece to be picked up by the manipulator’s end-effector is a rectangular parallelepiped. The images of the workpiece are captured from above by a CCD camera attached to the end-effector, as the end-effector moves toward the workpiece. The image of the workpiece’s top surface is a quadrangle. Only information about the quadrangle is of interest, so the captured RGB color image is first converted into YCbCr color space. The converted image is then preprocessed with the thresholding technique, and opening and closing operations to yield a clear binary image of the target area, as shown in Fig. 1. Finally, all the image features can be obtained from this target area.

## 3 Control strategy for picking up the target object

### 3.1 Selecting image features

Six image features are used to determine the translational and orientational motion in 3D. Each image feature can uniquely direct one D.O.F. of motion relative to the camera frame. Before the image features are defined, the coordinate symbols are provided. Specifically, as displayed in Fig. 2, let  $\delta^E X_1, \delta^E X_2$  and  $\delta^E X_3$ , represent differential changes in translation along the  $^E X, ^E Y$  and  $^E Z$  axes of the end-effector frame. In this study, the end-effector frame is fixed on the surface of the flange of the manipulator’s last link, with the  $^E Z$  axis along the last joint axis. Let  $\delta^E X_4, \delta^E X_5$  and  $\delta^E X_6$

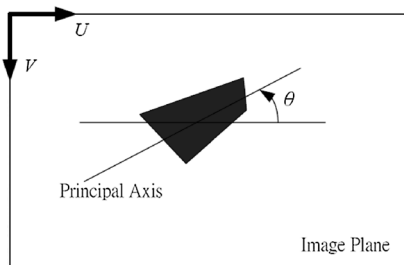


Fig. 1 Image of the workpiece’s top surface

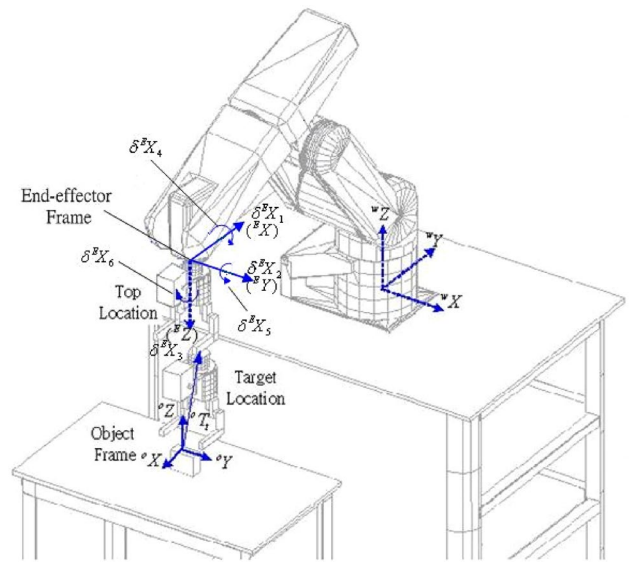


Fig. 2 End-effector and object frames

denote the differential changes in orientation about the  $^E X, ^E Y$  and  $^E Z$  axes of the end-effector frame. Furthermore, the camera frame must be considered. The origin of the camera frame is at the center of the camera, while its  $z$ -axis is the optical axis of the camera. Similarly,  $\delta^C X_i$  for  $i = 1, 2, \dots, 6$ , represent the differential movements with respect to the camera frame.

The two-dimensional geometric moment,  $m_{pq}$ , and central moment,  $\mu_{pq}$ , of order  $p + q$  for a  $(M \times N)$  discretized image,  $f(x, y)$ , are defined as

$$m_{pq} = \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} x^p y^q f(x, y), \tag{1}$$

$$\mu_{pq} = \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} (x - x_g)^p (y - y_g)^q f(x, y), \tag{2}$$

where  $(x_g, y_g)$  is the coordinate of image centroid. According to the definitions and the properties of image moments [5], we select the six image features as follows.

$F_1 = y_g = m_{01}/m_{00}$ , coordinate of the centroid of the quadrangular image. It directs camera motion  $^C X_1$ ;

$F_2 = x_g = m_{10}/m_{00}$ , coordinate of the centroid of the quadrangular image. It directs camera motion  $^C X_2$ ;

$F_3 = 1 - A_2/A_1$ , where  $A_i$  denotes the area of the quadrangle,  $m_{00}, i = 1$  in the desired pose and  $i = 2$  in the present pose.  $F_3$  guides camera motion  $^C X_3$ ;

$F_4 = f_y = (s_2 c_3 - c_2 s_3)/K$ , variation due to the rotation about the axis, which passes through the center of gravity of the quadrangle and is parallel to the  $V$ -axis in the image plane, where  $s_2 = \mu_{30} - 3\mu_{12}, c_3 = (\mu_{20} - \mu_{02})^2 - 4\mu_{11}^2, c_2 = \mu_{03} - 3\mu_{21}, s_3 = 4\mu_{11}(\mu_{20} - \mu_{02}), K = I_1 I_3^{3/2} / \sqrt{m_{00}}$ ,

$I_1 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2$ , and  $I_3 = \mu_{20} + \mu_{02}$ .  $F_4$  directs camera motion  ${}^C X_4$ ;

$F_5 = f_x = (c_2 c_3 + s_2 s_3)/K$ , variation due to the rotation about the axis, which passes through the center of gravity of the quadrangle and is parallel to the  $U$ -axis in the image plane. It directs camera motion  ${}^C X_5$ , and  $F_6 = 0.5 \tan^{-1} [2\mu_{11}/(\mu_{20} - \mu_{02})]$ , angle between the  $U$  axis and the principal axis of least inertia of the quadrangle in the image plane.  $F_6$  guides camera motion  ${}^C X_6$ .

### 3.2 Motion planning based on behavior design

In this work, fuzzy rules are used to enable the controllers to map image features in the image space onto motion commands in the camera space. These are then transformed to the commands in relation to the end-effector frame, which eventually control the manipulator.

#### 3.2.1 Six main behaviors

The complete manipulation task can be divided into two categories of behaviors—vision-based behaviors: Center 1, Center 2, Zoom, Yaw, Pitch and Roll, and non-vision-based behaviors: Top and Catch. The manipulation starts with the Top behavior to move the end-effector to the top location, where the pose of the end-effector in relation to the manipulator base is fixed. The first three vision-based behaviors are the translational motions of the camera toward the workpiece, while the other three vision-based behaviors are the orientational motions of the camera. The Catch is a non-vision-based behavior. Only when the end-effector has reached the target location is the Catch activated and moves the end-effector a short distance forward. The gripper then closes to grasp the workpiece. These behaviors are defined as follows:

Center 1 is based on the first image feature,  $F_1$ . This behavior translates the camera along the  ${}^C X$  axis of the camera frame to keep the centroid of the quadrangular image at the desired pixel in the image plane.

Center 2 is based on the second image feature,  $F_2$ . This behavior translates the camera along the  ${}^C Y$  axis of the camera frame to keep the centroid of the quadrangular image at the desired pixel in the image plane.

Zoom is based on  $F_3$ ; it moves the camera along the  ${}^C Z$  axis of the camera frame to keep the size of the object as a predefined value.

Yaw and Pitch are based on  $F_4$  and  $F_5$ , respectively; it rotates the camera about  ${}^C X$  and  ${}^C Y$  to keep the top surface of the target object parallel to the image plane.

Roll is based on  $F_6$ ; it rotates the camera about  ${}^C Z$  so that the principal angle equals that in the reference image, in which the gripper’s two fingers are arranged parallel to the short sides of target.

Vision-based behaviors are defined from the perspective of an eye-in-hand camera, so movements are performed relative to the camera frame. Figure 3 presents these behaviors.

#### 3.2.2 Neural-fuzzy controller

The best control law or membership functions of traditional fuzzy controllers can be determined by experience. However, the manipulation tasks are nonlinear and coupled. None set of membership functions is good for all of the work environment. With respect to learning ability of the artificial neural network, back-propagation architecture is the most popular and effective for solving complex and ill-defined problems. Hence, six simple NFCs (neural-fuzzy controllers) [6], using back-propagation, are designed herein. One image feature is input to each controller, which changes one D.O.F. of the camera motion as the output. The back-propagation algorithm is used only to adjust the consequents of fuzzy rules for each NFC at each iteration during the manipulation. Restated, the camera is guided intuitively by the image features on the image plane.

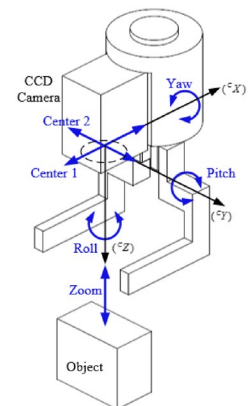
Fuzzy singleton rules are applied to simplify the NFCs; they are defined as follows:

$$R_i^j : \text{if } \delta F_i \text{ is } A_i^j, \text{ then } \delta {}^C X_i \text{ is } w_i^j, \tag{3}$$

where input variable  $\delta F_i$  is an image feature error; output variable  $\delta {}^C X_i$  denotes a relative motion command in the camera frame;  $A_i^j$  are linguistic terms of the precondition part with membership functions  $\mu_{A_i^j}(\delta F_i)$ , and  $w_i^j$  represent

the real numbers of the consequent part,  $i = 1, 2, \dots, 6$  and  $j = 1, 2, \dots, 7$ . That is, each  $i = 1, 2, \dots, 6$ , can be treated as a NFC with seven rules to control one D.O.F. of motion relative to the camera frame. The input membership functions and the singletons of the consequent parts of fuzzy rules for each NFC are shown in Fig. 4. Herein, a simplified defuzzifier is used. The final output  $\delta {}^C X_i$  of the neural-fuzzy system is determined according to  $\delta {}^C X_i = \sum_{j=1}^7 \mu_{A_i^j} w_i^j$ .

Fig. 3 Movements associated with defined behaviors



In this work, only real numbers  $w_i^j$  are tuned on-line. Accordingly, the error function to be minimized is defined by

$$E_i = \frac{1}{2}(F_i^r - F_i)^2 = \frac{1}{2}(\delta F_i)^2 \tag{4}$$

The derivative of an error function  $E_i$  with respect to the  $j$ th consequent can be obtained as:

$$\frac{\partial E_i}{\partial w_i^j} = \frac{\partial E_i}{\partial(\delta^C X_i)} \frac{\partial(\delta^C X_i)}{\partial w_i^j} \approx \frac{E_i(t) - E_i(t-1)}{\delta^C X_i(t) - \delta^C X_i(t-1)} \frac{\partial(\delta^C X_i)}{\partial w_i^j} \tag{5}$$

To decrease  $E_i$  with respect to  $w_i^j$ , the consequent changes at time stage  $t$ , and  $\Delta w_i^j(t)$  can be chosen as  $\Delta w_i^j(t) = -\eta \partial E_i / \partial w_i^j$ , where  $\eta$  is the learning-rate parameter.

Additionally, for the fast learning speed, learning rate [7] is designed as saturated linear function as follows:

$$\eta_i(\delta F_i) = \begin{cases} (\eta_i)_{\max}, & \text{if } |\delta F_i| > (\delta F_i)_{\text{sat}} \\ \frac{(\eta_i)_{\max} - (\eta_i)_{\min}}{(\delta F_i)_{\text{sat}}} \delta F_i + (\eta_i)_{\min}, & \text{if } |\delta F_i| < (\delta F_i)_{\text{sat}} \end{cases} \tag{6}$$

where  $(\eta_i)_{\min}$  and  $(\eta_i)_{\max}$  are, respectively, minimum and maximum learning rate that can be chosen by the user, and  $(\delta F_i)_{\text{sat}}$  is saturated difference value that can also be chosen by the user. Thus, a learning rule for adapting the consequent at time stage  $t$  can be given as  $w_i^j(t+1) = w_i^j(t) + \Delta w_i^j(t)$ .

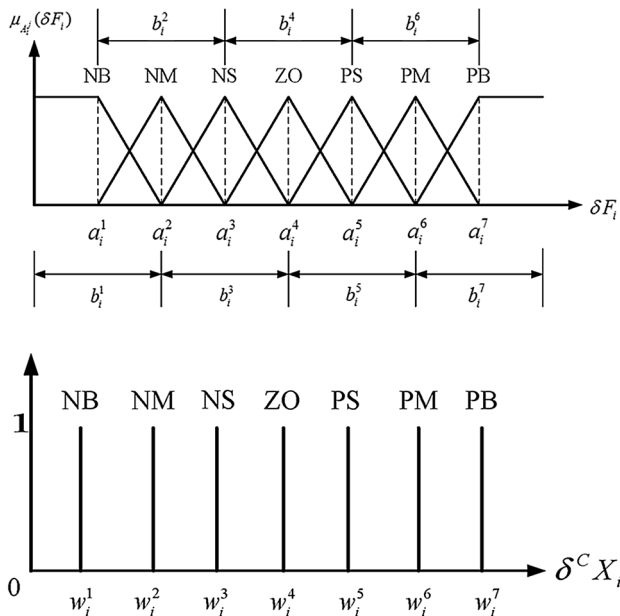


Fig. 4 Membership functions of the input/output variables

### 3.3 Rough motion transformation

In this work, the pose of the camera in relation to the end-effector is invariant, as shown in Fig. 5(a), so the camera and the end-effector can be regarded as a rigid body. After the motion of the rigid body has been analyzed, the transformation from the output values in relation to the camera frame to the motion commands with respect to the end-effector frame is obtained.

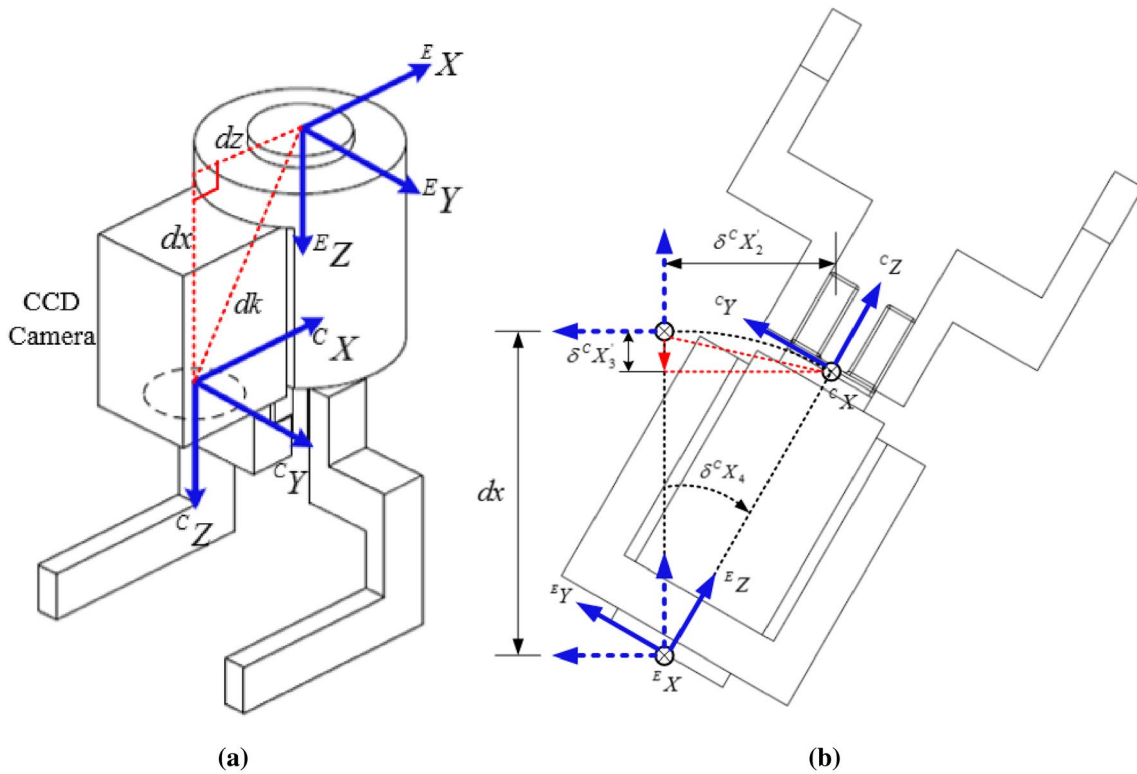
Figure 5(b) demonstrates that if the controller output is only a rotation  $\delta^C X_4$  about  ${}^C X$  but the command sent to the manipulator is a rotation about  ${}^E X$ , then the unexpected camera displacements in the  $-{}^C Y$  and  $-{}^C Z$  directions are  $\delta^C X'_2$  and  $\delta^C X'_3$ , respectively. Similar situations occur in  $\delta^C X_5$  and  $\delta^C X_6$ . Accordingly, the motion command sent to the manipulator should be transformed as:

$$\begin{aligned} \delta^E X_1 &= \delta^C X_1 - dz \cdot \sin(\delta^C X_6) \cdot \tan(\delta^C X_6/2) - 2dk \\ &\quad \cdot \sin(\delta^C X_5/2) \cdot \cos(90^\circ - \delta^C X_5/2 - \tan^{-1}(dz/dx)) \\ \delta^E X_2 &= \delta^C X_2 + dz \cdot \sin(\delta^C X_6) + dx \cdot \sin(\delta^C X_4) \\ \delta^E X_3 &= \delta^C X_3 + dx \cdot \sin(\delta^C X_4) \cdot \tan(\delta^C X_4/2) - 2dk \\ &\quad \cdot \sin(\delta^C X_5/2) \cdot \sin(90^\circ - \delta^C X_5/2 - \tan^{-1}(dz/dx)) \\ \delta^E X_4 &= \delta^C X_4, \quad \delta^E X_5 = \delta^C X_5, \quad \delta^E X_6 = \delta^C X_6, \end{aligned} \tag{7}$$

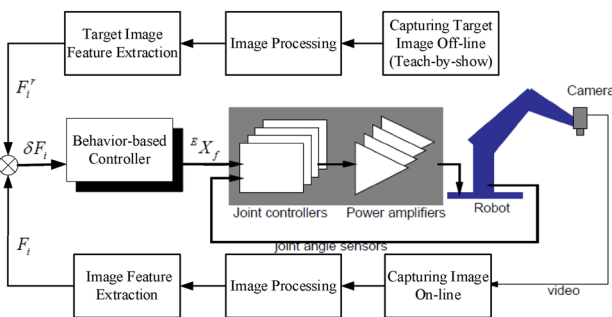
where  $dx$  is the distance between the axes  ${}^C X$  and  ${}^E X$ , and  $dz$  represents the distance between the axes  ${}^C Z$  and  ${}^E Z$ . The two lines associated with  $dx$  and  $dz$  are assumed to be mutually perpendicular. They are measured roughly using a ruler and the naked eye. Therefore,  $dk = \sqrt{(dx)^2 + (dz)^2}$  is assumed.

### 3.4 Control strategy

The behavior-based look-and-move control structure is depicted in Fig. 6. The manipulation starts with the Top behavior to move the end-effector to the top location, as depicted in Fig. 2, to perform pick-and-place tasks.  $F_i^r$  is the value of  $F_i$  measured by the teach-by-showing method. This reference image is captured by the vision system when the end-effector is driven by a teaching box to the target location,  ${}^o T_r$ , relative to the object frame, as shown in Fig. 2, where the object frame is fixed on the top surface of the object. The target location is defined as follows. The end-effector is initially driven by a teaching box to a location that allows the gripper to grasp the workpiece. Then, the end-effector is driven to another location, which is a safe distance from the preceding location (in this study, 10 cm above it). This “another location” is called the “target location”. Consequently, the reference features that correspond to the reference image at the target location are  $F_1^r, F_2^r, F_3^r, F_4^r, F_5^r$  and  $F_6^r$ .



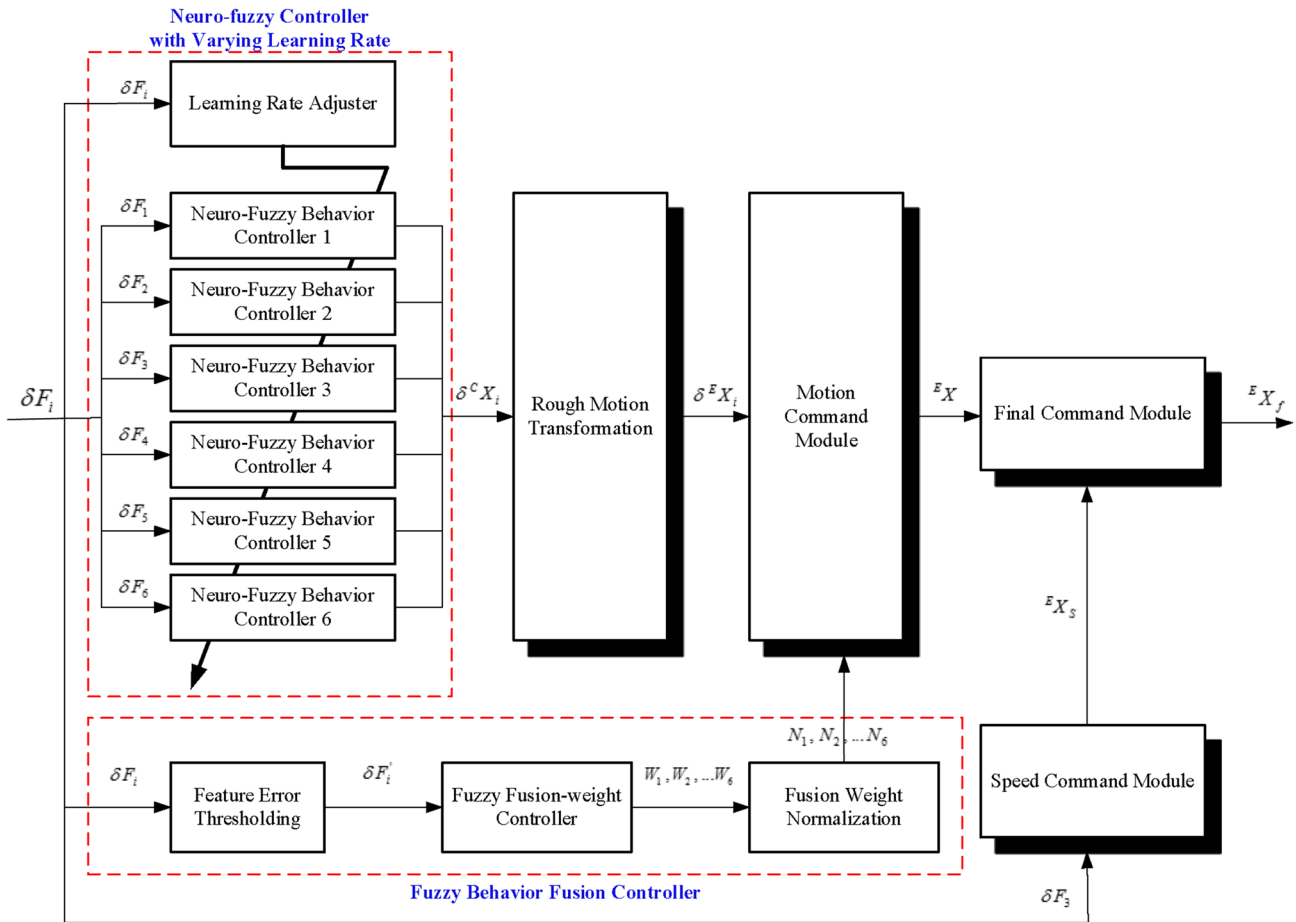
**Fig. 5** (a) End-effector and camera frames, (b) unexpected camera displacement caused by a rotation about  $E_X$



**Fig. 6** Behavior-based look-and-move control structure

Each  $\delta F_i$ , for  $i = 1, 2, \dots, 6$ , defined as the error between  $F_i$  and the feature value of the reference image at the target location,  $F'_i$ , is then input to the proposed behavior-based controller to obtain the motion command in the end-effector frame. The internal structure of the behavior-based controller is shown in Fig. 7. The relative motion command in the camera frame generated from the NFCs can be represented as a six-element vector  $[\delta^c X_1, \delta^c X_2, \delta^c X_3, \delta^c X_4, \delta^c X_5, \delta^c X_6]^T$ . The rough motion transformation can be applied to transform the relative motion command in the camera frame to the relative motion command  $[\delta^E X_1, \delta^E X_2, \delta^E X_3, \delta^E X_4, \delta^E X_5, \delta^E X_6]^T$  in the end-effector frame.

Each element of the original relative motion command in the camera frame is independent of each other. However, most elements of the relative motion command in the end-effector frame are coupled. To achieve the smoother manipulation, these behaviors in the end-effector frame are then fused to produce a final command action to reach the target location using the proposed behavior fusion scheme. The inputs to the behavior fusion controller are image feature errors,  $\delta F_i$ , for  $i = 1, 2, \dots, 6$ , while the outputs are the normalized fusion weights of the behaviors. Before input to the fuzzy fusion-weight controller, the image feature error,  $\delta F_i$ , is compared with the preset threshold to decide whether the corresponding behavior needs to be fused. The membership functions of the input and output variables are displayed in Fig. 8. According to the membership functions of the input variables, the rule base of the fusion-weight controller consists of seven rules. At each time stage, the input variables are first fuzzified and the fuzzy inference is made. The crisp value of the behavior fusion weight,  $W_i$ , is then determined by the center of gravity defuzzification. Finally, the defuzzified weights are normalized to be  $N_i = W_i / \sum W_i$ . Accordingly, the fused relative motion command becomes  $E X = [E X_i]_{6 \times 1} = [\delta^E X_i \cdot N_i]_{6 \times 1}$  in the end-effector frame. In the speed command module, the motion speed of the manipulator is dynamically adjusted and set as some percentage



**Fig. 7** Internal structure of behavior-based controller

(10–40%) of the full speed based on the third image feature error,  $\delta F_3$ .

During the manipulation, the non-vision-based behavior, Top, is initially activated to move the end-effector of the manipulator to the top location. Whenever  $|\delta F_i|$  is bigger than the specified limiting value,  $\epsilon_i$ , the corresponding vision-based behaviors are then activated and fused to command the end-effector to approach the target location. This process is iteratively performed until all the image feature errors are below the specified limiting values. Finally, the non-vision-based behavior Catch is inspired, and the end-effector is commanded to move through a short distance (in this study, 10 cm) along the  $EZ$  axis of the end-effector frame to grasp the workpiece.

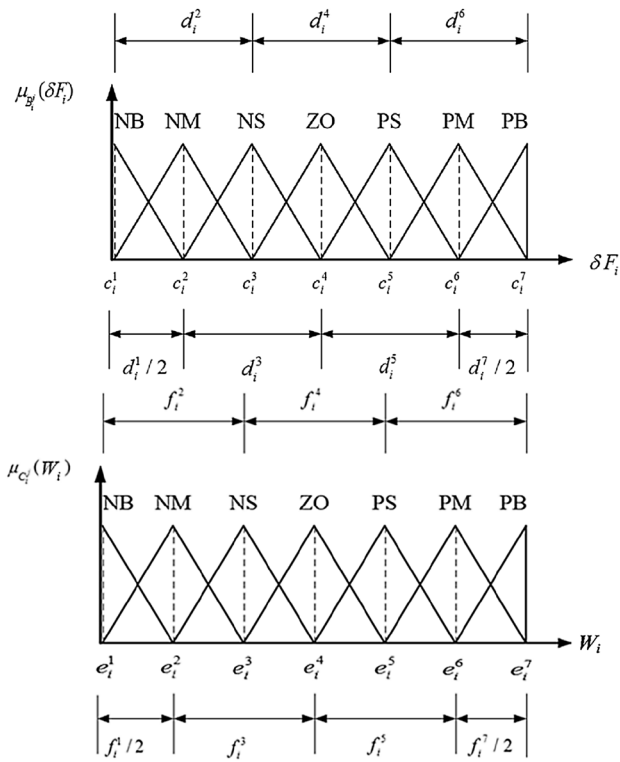
### 4 Experimentation

The mobile base is stopped and fixed next to the workstation, which is roughly parallel to the surface of the workstation to verify the proposed control strategy. As presented in Fig. 9,

the workpiece is pointed in three directions and placed in six different positions, which are separated by 15 cm, to simulate the possible orientation and position errors that arise in the application stage. In each location, the workpiece is tilted by  $15^\circ$  and  $25^\circ$  to the station surface to simulate non-flat ground in the application stage.

In experiments performed to evaluate the positioning performance of the eye-in-hand manipulator, the end-effector of the manipulator is firstly driven to the top location. Then, the end-effector is visually guided to grasp the workpiece according to the proposed control strategy with the preset parameters. In the approaching stage, the image feature errors drop as the number of execution steps grows. This stage continues until the errors in the image features,  $\delta F_i$ , for  $i = 1, 2, \dots, 6$ , are below  $\epsilon_1 = 2, \epsilon_2 = 2, \epsilon_3 = 5, \epsilon_4 = 0.001, \epsilon_5 = 0.001$  and  $\epsilon_6 = 0$ , respectively. Figure 10 displays the images in the course of approaching the workpiece.

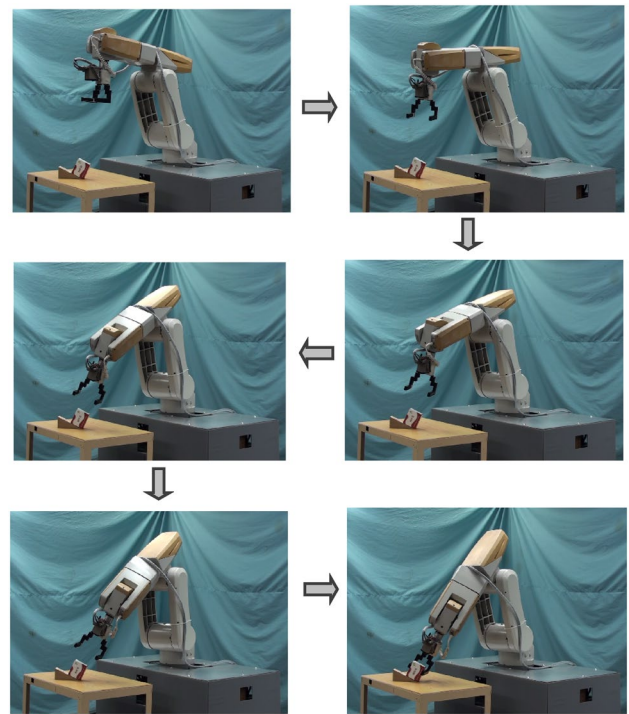
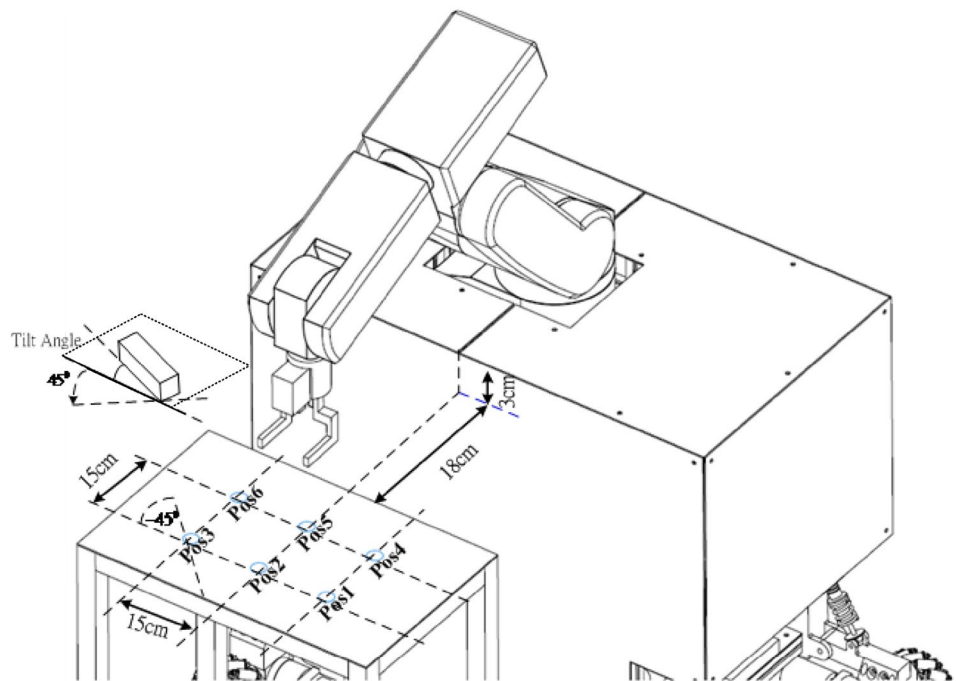
The final and desired locations of the end-effector are recorded in the task manipulation. The coordinate transformation matrix between those locations of the end-effector can then be determined. Consequently, the position error



**Fig. 8** Membership functions of the input/output variables

can be defined by the magnitude of the position vector of the matrix. The orientation error can be defined as the angle of rotation about the principle axis, obtained from the rotational transformation of the matrix.

**Fig. 9** Possible locations of the workpiece to be picked up



**Fig. 10** Images in the course of approaching the workpiece in Pos2 with a rotation of 45° and a tilt of 25°

Table 1 shows the resulting positioning errors and the number of steps for each test. The number of steps is that required for the end-effector to travel from the top location

**Table 1** Positioning errors of the eye-in-hand manipulator and the number of steps for each test

Workpiece location	Tilt	Position error (mm)	Orientation error (°)	Number of steps
<b>Pos1</b>				
No rotation	0°	1.1422	0.3453	10
	15°	1.0735	0.5122	18
	25°	2.3574	0.7245	20
Rotation 45°	0°	1.3326	0.5365	9
	15°	0.9337	1.0238	21
	25°	1.9926	0.3856	23
Rotation -45°	0°	1.0214	0.5771	11
	15°	1.7524	1.1345	18
	25°	2.1013	0.8852	24
<b>Pos2</b>				
No rotation	0°	0.5961	0.4195	9
	15°	1.5521	0.3773	17
	25°	2.3118	0.5370	20
Rotation 45°	0°	1.0477	0.1189	11
	15°	1.0784	1.0051	20
	25°	2.0270	1.0623	22
Rotation -45°	0°	0.8750	0.7575	9
	15°	0.8821	0.9010	20
	25°	1.9598	1.0190	23
<b>Pos3</b>				
No rotation	0°	0.4575	0.5474	12
	15°	1.2010	0.4851	18
	25°	2.3903	0.8013	20
Rotation 45°	0°	1.0103	1.1162	10
	15°	1.7310	0.7629	19
	25°	2.1531	0.4365	23
Rotation -45°	0°	0.9050	0.4887	11
	15°	1.1877	1.1043	20
	25°	2.4551	0.8032	25
<b>Pos4</b>				
No rotation	0°	1.2060	0.5570	10
	15°	1.5124	0.4241	18
	25°	2.0121	0.5913	20
Rotation 45°	0°	0.9865	1.2011	11
	15°	0.8258	1.0022	19
	25°	1.9547	0.6804	24
Rotation -45°	0°	1.1003	0.8213	12
	15°	1.8332	1.0470	21
	25°	2.3958	0.7749	24
<b>Pos5</b>				
No rotation	0°	0.7197	0.5903	9
	15°	1.3210	1.0355	17
	25°	2.1489	0.9110	21
Rotation 45°	0°	0.8412	0.7553	10
	15°	1.3189	0.7029	20
	25°	1.9981	0.8454	23

**Table 1** (continued)

Workpiece location	Tilt	Position error (mm)	Orientation error (°)	Number of steps
Rotation -45°	0°	1.0602	1.0030	9
	15°	2.0139	0.6574	20
	25°	2.2591	0.8059	24
<b>Pos6</b>				
No rotation	0°	0.8015	1.0560	9
	15°	1.0354	0.6523	18
	25°	2.1052	0.4321	21
Rotation 45°	0°	1.1851	0.5770	10
	15°	0.9881	1.1066	20
	25°	2.4043	0.8721	23
Rotation -45°	0°	1.1003	0.7664	11
	15°	1.4576	0.8330	21
	25°	2.2773	0.5654	25

to the target location. Executing this behavior-based control strategy takes a minimum of about 9 steps and a maximum of 12 steps in cases in which the workpiece is arranged without a tilt. The initial orientation of the workpiece only weakly influences the total execution time. In the cases in which the tilt angle is 15° or 25°, about 17 steps to 25 steps are required. As the tilt angles are increased, more steps are needed. In all of the tests, the final position error is less than 2.46 mm, and the final orientation error is less than 1.21°.

### 5 Conclusion

This paper adopts an uncalibrated eye-in-hand vision system to provide visual information to the mobile manipulator to pick up a workpiece located on the station. A novel vision-guided control strategy with a behavior-based look-and-move structure is proposed. This strategy includes the NFCs with varying learning rate, the rough motion transformation and the behavior fusion scheme. Notably, this rough motion transformation is inaccurate. However, the designed NFCs handle the inaccuracy by tuning the consequents of the fuzzy rules using the back-propagation algorithm. This process saves considerable time without the extensive computation of hand-eye calibration. Finally, the proposed control strategy is experimentally applied to realize a manipulator that can fast approach a target object and precisely position its end-effector in the desired relative pose to the object, independently of where the target object is located on a station. Significantly, the superiority of this research over the conventional one can be shown as follows: (1) No camera calibration and hand/eye calibration are performed. (2) The selected image features are not sensitive to illumination and the distance between the camera and the workpiece. (3) The



end-effector of the manipulator is controlled to approach the workpiece more smoothly and faster by the behavior fusion scheme and the speed command module of the proposed control strategy.

**Acknowledgements** The authors would like to thank the National Science Council of the Republic of China, for financially supporting this research under Contract No. NSC 102-2221-E-006-295.

## References

1. Hvilshøj M, Bøgh S, Nielsen OS et al (2012) Autonomous industrial mobile manipulation (AIMM): past, present and future. *Ind Robot* 39(2):120–135
2. Hutchinson S, Hager GD, Corke PI (1996) A tutorial on visual servo control. *IEEE Trans Robot Autom* 12(5):651–670
3. Wasik Z, Saffiotti A (2003) A hierarchical behavior-based approach to manipulation tasks. In: *Proceedings of the IEEE international conference on robotics and automation*, pp. 2780–2785
4. Wasik Z, Saffiotti A (2002) A fuzzy behavior-based control system for manipulation. In: *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems*, pp. 1596–1601
5. Chaumette F (2004) Image moments: a general and useful set of features for visual servoing. *IEEE Trans Robot* 20(4):713–723
6. Suh IH, Kim TW (1994) Fuzzy membership function based neural networks with applications to the visual servoing of robot manipulators. *IEEE Trans Fuzzy Syst* 2(3):203–220
7. Kim TW, Yuh J (2005) Fast on-line neuro-fuzzy controller for autonomous underwater vehicles. In: *Proceedings of the international offshore and polar engineering conference*, pp. 289–294