February, 2017
Thesis for Master's Degree

A study on the evaluation of the developed robot vision control schemes in real time for the estimation of the fixed target

Dept. of Mechanical Engineering

The Graduate school of Chosun University

MINWOO JANG

A study on the evaluation of the developed robot vision control schemes in real time for the estimation of the fixed target

February 24, 2017

Dept. of Mechanical Engineering

The Graduate school of Chosun University

MINWOO JANG

A study on the evaluation of the developed robot vision control schemes in real time for the estimation of the fixed target

Advisor : Prof. Wan Shik Jang, Ph.D.

A Dissertation Submitted to the Departure of Mechanical Engineering and the Graduate School of Chosun University in partial fulfillment of the requirement for the degree of Master of Engineering

October, 2016

Dept. of Mechanical Engineering

The Graduate school of Chosun University

MINWOO JANG

This certifies that the dissertation
of MINWOO JANG is approved

Thesis Supervisor	Chosun University	Prof.	Sang Hwa Jeong
Committee Member	Chosun University	Prof.	Wan Shik Jang
Committee Member	Chosun University	Prof.	Yoon Gyung Sung

November, 2016

The Graduate school

Chosun University

TABLE OF CONTENTS

Chapter 3. Mathematical modeling for parameter estimation

Chapter 4. Robot's vision control scheme for fixed target estimation

List of Photos

List of Figures

List of Tables

NOMENCLATURE

${}_{i}^{i-1}T$	Transfer matrix from frame $\{i-1\}$ to frame $\{i\}$
P^{i}	Position vector of the cue at the tool frame
$\overline{F}$	Position vector of the cue at the robot base frame
C_k	View parameter
X_m	X component of estimation model
Y_m	Y component of estimation model
F_x	$\hat{X}$ component of position vector $\overline{F}$
F_y	$\hat{Y}$ component of position vector $\overline{F}$
F_z	$\hat{Z}$ component of position vector $\overline{F}$
$J(C_k)$	Performance index in parameter C
X_c^i	X component of vision data for acquired at step i
Y_c^i	Y component of vision data for acquired at step i
X_m^i	X component of estimation model for computed at step i
Y_m^i	Y component of estimation model for computed at step i
ΔC	Parameter correction vector
A	Matrix of $(2\times i\times j)\times(6)$
A^T	Transpose of matrix A
W	Weighting matrix
$J(\theta_i)$	Performance index in joint angle
$\Delta\theta$	Joint angle correction vector
R	Residual vector
B	Matrix of $(2\times q\times j)\times(4)$
B^T	Transpose of matrix B

ABSTRACT

A Study on the Evaluation of the Developed Robot Vision Control Schemes in Real time for the Estimation of the Fixed Target

Jang, Min Woo

Advisor : Prof. Jang, Wan Shik Ph.D.

Department of Mechanical Engineering,

Graduate School of Chosun University

산업 현장에 등장한 산업용 로봇은 초기에는 단순 반복 작업에 이용된 이후 다양한 작업환경에 대응하기 위해 인간의 지능 중 가장 중요한 요소인 시각기능을 로봇에 탑재하여 자율성을 확보한 로봇 비젼 시스템에 대한 연구가 활발히 진행되고 있으며, 로봇에 비젼 센서를 이용하여 로봇의 자율적인 주행경로 결정 기술, 2차원 시각정보의 3차원 공간 재현, 물체 추적기술 등에 대한 많은 연구가 수행되고 있다.

현재 진행되고 있는 대부분 연구들의 문제점은 로봇에 비젼시스템을 적용할 때에 가장 중요한 요소인 카메라의 위치, 방향 및 초점거리에 대한 보정이 정확하게 되어 있지 않으면 로봇은 정상적인 위치 보정을 하지 못하고 오동작을 할 우려가 있으며, 특히, 카메라와 로봇 사이 상대적인 위치와 카메라 방위 및 초점거리가 변하면 이에 대한 카메라 보정계수를 다시 계산해야 할 번거로움이 있다.

이리하여, 본 연구는 2-D 카메라 이미지 평면에서 물체 정보와 3-D 공간상 로봇 위치 정보를 변환함과 동시에, 카메라 및 로봇 위치 변화에 대해 카메라 매개변수를 능동적으로 조절 할 수 있는 비젼시스템 모델을 사용하였다. 제시된 로봇 비젼 시스템 모델을 이용하여 로봇이 타겟을 향해 이동하는 동안 얻어진 비젼데이터를 효율적으로 처리하기 위해서 N-R 방법을 적용한 로봇 비젼 제어기법과 EKF 방법을 적용한 로봇 비젼 제어기법을 제시하고자 한다. 특히, N-R 방법의 로봇 비젼 제어기법은 최근 데이터에 비중을 두는 가중치 행렬을 적용한 경우와 적용하지 않은 경우로 구분하였다. 최종적으로, 제시된 3 개 실시간 로봇 비젼 제어기법들을 사용하여 강체 배치 작업을 위한 공간상 위치가 알려지지 않은

고정 타겟 추정 실험을 수행하여 제시된 제어기법의 정밀도와 처리시간을 비교 분석하여 효율성을 파악하고자 한다.

Chapter 1. Introduction

1.1 Research background

Robots have been developed for the purpose of replacing simple repetitive work or in 3D industries by humans in industrial fields. Recently, owing to the development of robot related technology, an intelligent robot that recognizes the external environment and judges the situation by itself has emerged. Therefore, the field of robotics has penetrated a wide range of applications, and now robots are becoming a part of human life, coexisting with humans. Robots are used in various applications that include industrial applications, domestic applications such as cleaning and housekeeping, medical applications such as surgery and health care, defense applications such as robot soldiers and unmanned drones, agricultural applications such as agricultural crop harvesting, educational applications such as assembly kits and training robots, and transportation applications such as unmanned vehicles, traffic control.

Among the various fields, the latest technology trends in industrial robots involve using high-speed CPU visual sensors, 2-D and 3-D image processing, visual technology related to computer aided design(CAD) matching, fast and accurate work by six axis sensor controls, automation of precision assembly work, and collision detection. Especially, numerous technological advances have also been made in the field of force technology, including hand and variable strokes, and hand and safety technologies related to gripping force control ability.

For robots to be used in industrial fields, where robots are required to perform several tasks, various abilities must be provided. As robots need to be independent, they must be able to judge the environment around themselves, be able to control their own systems, and be able to carry out their tasks. To perform such tasks, robots must have artificial intelligence. Image recognition is

an important factor among the various artificial intelligence components that robots should possess. So far, research has been conducted on machine vision that combines image processing technology with different machinery for image recognition. Furthermore, much research has been done on robot vision using machine vision technology. First, the research trends in image processing are as follows.

Various methods have been proposed for extracting object information from image information as the demand for systems that perform tasks using object information extracted from images increases.
Feddema[1] presented a method for visually tracking moving objects using a single moving camera. Assa[2] proposed a sensor fusion technique that estimates the object shape in real time with high accuracy and precision by complementing the multi-camera sensor fusion method estimating only a few number of objects among various methods of estimating the proposed object based on a single camera. In addition, Lowe[3] proposed a method for fitting both parameters and models inside an image on a curvature, and studied the criteria for fitting 2-D images and 3-D models in most vision-based programs.

As mentioned above, machine vision technology, which analyzes images using a computer and an optical system such as a camera and lens, in the fields of machinery and agriculture, as well as image processing technology, is a new and useful field in the sense that it is a substitute for human judgment. In recent years, research on machine vision systems that combines image processing technology with advanced inspection equipment is actively being studied.

Jahari[4] introduced a system for optimizing performance by monitoring the quality of harvested crops in real time using the developed dual illumination method and mounting system above the entrance of the harvester for evaluation. Additionally, Huang[5] developed a machine vision system for

detecting defects in internal atomizing nozzles using neural networks and image processing techniques. This system evaluated geometric features using color features involving gray level differences and differences in diameter and distance.

In recent years, the plant have been aiming to produce small quantities of various products in complex various industrial fields. Therefore, there are using a robotic vision beyond the limited machine vision, and research in this regard is actively being conducted.

Hosoda[6] used a vision sensor on a legged robot to create a servoing system consisting of a controller and a visual servoing controller that maintains constant distances without pre-programming leg motion in the controller, a study was conducted to control the posture by tracking the shaking object. Choe[7] proposed a visual servo control method that uses vision to track the hole of a bolt using the automation robot in the construction field. In addition, Asl[8] proposed a control system using an image-based visual servoing(IBVS) method to solve the problem of vision-based controller design for translational motion and yaw rotation of unmanned aerial robots. Based on this system, the effectiveness of the presented control system was verified through the dynamic characteristics of the images obtained from the perspective image moments in the virtual image plane.

Recently, Oh[9] proposed a 2-D camera-based object recognition and robotic arm control method, which is a coordinate system conversion method. This method used a circular Hough transformation to obtain the angle of each joint of the robotic arm in real space from the object information obtained from the 2-D camera image. In addition, Kim[10] introduced GP technology based on a color detection model for object detection of humanoid robot vision, and proposed a non-parametric multi-color detection model using the evolution of GP.

Several methods have been proposed to develop the robot vision algorithm. The Newton-Raphson(N-R) method and the Extended Kalman Filter(EKF) method, which are the numerical control schemes to be implemented in this paper, have opposite advantages and disadvantages. So far, the research trends of these methods are as follows.

1.1.1 Research trends in the N-R Method

Several researchers have conducted general studies of the N-R method, which is an iterative technique, and its development for research applied to robot control. The main trends of general research on the N-R method include the following. Choi[11] studied the method of reducing the computation speed and the maximum error rate by storing the initial approximate solution in a table based on the geometric mean in the Newton-Raphson algorithm. This is important in evaluating the overall performance based on determination of the initial approximate solution. Bae[12] proposed a slot parameter extraction method to design a wide-surface resonance-type waveguide antenna using the Newton -Raphson method as a numerical solution to solve nonlinear equations, and developed a program to calculate slot parameters based on a formula proposed by Elliott for wave plate antenna radiation plate design theory.

Several previous studies were performed by applying the fore-mentioned research results to the development of robot control schemes. Durmus[13] used the Newton-Raphson method to minimize the cost function of a generalized predictor that represented the error between the reference trajectory and the actual trajectory of the robot in a six-axis robot manipulator. Through this study, the Newton-Rahpson's minimization method and generalized prediction control algorithm were derived. Yang[14] proposed a Global Newton-Raphson method that allowed parallel robots to directly obtain answers in real-time without the need for a parallel robot to analyze the 6 degrees of freedom of

kinematics. Additionally, the parallel robots generated linear motion with three degrees of freedom as well as angular motion of three degrees of freedom and could feedback the actuator position measured by the position sensor in real time.

Recently, research on the robot vision control scheme involved the application of the weighting matrix to the Newton-Rahpson method to improve the accuracy. Jung[15] proposed an autonomous navigation intelligent system that applied a new weighting matrix to an algorithm, which could be performed in real time, when given a destination without any additional routing process or processing for control input calculation. The proposed algorithm was verified through simulation. Additionally, Kim[16] studied the effectiveness of template matching in various lighting conditions using a weighting matrix to use a vision-based object detection system in an underwater environment with noise or light limitations.

1.1.2 Research trends in the EKF Method

Since 2000, general research on the EKF method, which is recursive technique, and robot control schemes based on the general EKF method have steadily progressed.

The main trends of general research on the EKF method include the following. Rigatos[17] studied a sensorless control for DC and induction motors by estimating the angular velocity of a motor using the EKF method to measure the angle of the rotor. Karasalo[18] proposed an optimization-based adaptive Kalman filtering method to generate an estimation of a process noise covariance matrix by solving the data optimization problem on a short screen.

Several researchers applied the EKF method to the robot control method to improve processing speed. Chen[19] provided a camera calibration method that could be used the robot vision system for the motion control of a robot arm in a situation in

which the reference coordinate system and the camera coordinate system could not be accurately obtained by estimating the camera parameters using the EKF method.

Recently, studies controlling a mobile robot using the EKF method are actively conducted both domestically and abroad. Hwang[20] studied a vision tracking system that estimated the position of a robot using a slip detector and two Kalman filters because wheel and encoder data could not be relied in the event of sliding of the mobile robot. Darabi[21] used an Extended Kalman filter and a Sigma Point Kalman filter to perform simultaneous localization and mapping, which are among the most basic and challenging problems of a mobile robot equipped with a moving stereo vision sensor in an indoor environment. This was followed by reviewing and improving the estimated state of simultaneous localization and mapping by repeating the filter. Furthermore, Jeon[22] fused two methods by applying the EKF method to estimate the position using the vision system and the marker and applying the encoder on the mobile robot wheel. The study then applied the vector field histogram scheme to examine a method of avoiding obstacles in real time at a desired destination.

1.2 Research needs and objectives

Robots introduced for industrial applications in the 1980s were used for simple repetitive tasks. Since then, research on robot vision systems with autonomy secured by endowing visual ability, which is the most important factor of human intelligence, is actively conducted to cope with various working environments. Many studies on autonomous travel path determination technologies, 3-D space transform of 2-D visual information, and object tracking technologies have been conducted[23,24].

In particular, the problem of slow processing speed due to large amounts of data processing when applying vision systems to robots is solved through the steady development of large-scale memory devices such as large-scale integrate(LSI) devices since the late 1970s. In addition, binarization schemes or cue-based methods have been studied to minimize data processing times by minimizing the amount of data to be acquired by considering work conditions and characteristics[25,26]. With these technological advancements, in recent years, research is actively being conducted on applying vision systems to robots in various complex industrial fields. However, most of the ongoing research has the following problems. The compensation coefficients of position, orientation, and focal length of the camera are most important when applying a vision system to a robot. If these are not done correctly, the robot cannot perform normal position control, and can operation incorrectly. In particular, if the relative position between the camera and the robot, and the orientation and the focal length of the camera are changed, the compensation coefficients of the camera must be recalculated.

To solve these problems, this study uses a vision system model involving 6 camera parameters. These parameters combine in this model that relates object information in the 2-D camera image plane to object information in 3-D space. Specifically, the parameters can actively adjust the change of the camera and robot positions without calibration. Thus, based on this vision system, robot vision control schemes using the N-R method and EKF method are presented

to efficiently process vision data obtained while the robot moves towards the fixed target. In particular, the robot vision control scheme of the N-R method is classified into two cases, where one case is with the weighting matrix and the other case is without the weighting matrix, in order to examine the effect of data processing method.

Finally, the practicality of the three proposed control schemes is demonstrated by comparing the accuracy and processing time of the fixed target estimation through experiments of rigid body placement.

1.3 Research contents

This study involves two main topics. First, robot inverse kinematic model is developed to calculate robot joint angles when spatial coordinate values for a specified target are known. Second, three control schemes are proposed to estimate the position of the target using only vision data obtained through a camera when the spatial coordinate value of the specified target is unknown. The estimated values of the proposed control schemes are compared with the obtained values from the robot kinematic analysis to evaluate the effectiveness of the proposed control scheme. A test model with four cues is attached to the end of the robot to show the effectiveness of the proposed control technique. Additionally, the three cameras used in the study are placed on the image plane such that the test model did not deviate from the camera image plane and filled the camera image plane while the robot moved along the random motion trajectory towards the fixed target.

The contents of this thesis include the following. Chapter 2 presents the robot kinematic model and the vision system model. The actual joint angles of the robot for specified targets are calculated using the forward kinematic model and inverse kinematic models described in section 2.1 and section 2.2. Additionally, section 2.3 presents a vision system model that includes six parameters to describe the relative positions of cameras and robots, including uncertainties in camera orientation and focal length of camera. Chapter 3 presents the mathematical modeling of the N-R method and the EKF method to estimate camera parameters and robot joint angles based on the vision system model presented in section 2.3. In chapter 4, three robotic vision control schemes are proposed using the N-R method without the weighting matrix, N-R method with weighting matrix, and EKF method to estimate rigid fixed targets. Chapter 5 shows the test model of a rigid body using four cues without recognizing the entire object shape to improve the processing speed by minimizing the amount of data. Furthermore, ten initial stages and thirty robot movement stages are set for fixed target estimation for real-time rigid body placement task.

In chapter 6, the suitability of the vision system model is verified using the presented control schemes. In chapter 7, the positional values for the fixed target are calculated using four schemes such as the robot kinematics analysis, N-R method without the weighting matrix, N-R method with the weighting matrix, and EKF method. The structure of thesis involving research contents and method mentioned above is shown in Fig. 1-1

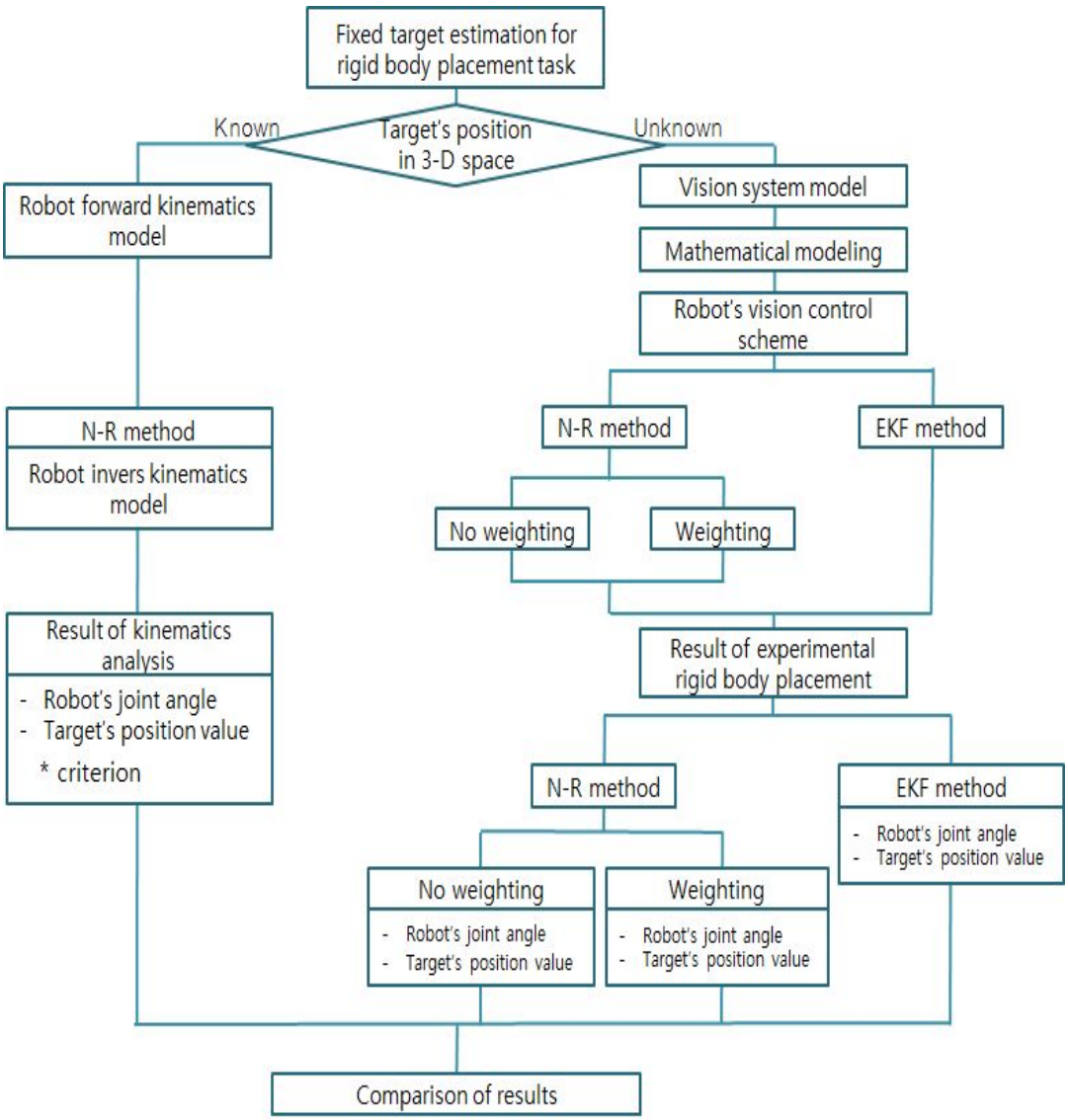


Fig. 1-1 Structure of thesis

Chapter 2. Robot kinematics model and vision system model

2.1 Robot forward kinematics

Robot forward kinematics involves calculating the end-effector position vector of the robot with respect to the robot base coordinate system when the joint angles of the robot are given. This model is applied to the vision system model in section 2.3.

Fig. 2-1 shows the link frame and link parameters of four joint axis for the Samsung SM7 four-axis scara type robot used in this study. The measured robot link parameters are listed in Table 2-1.

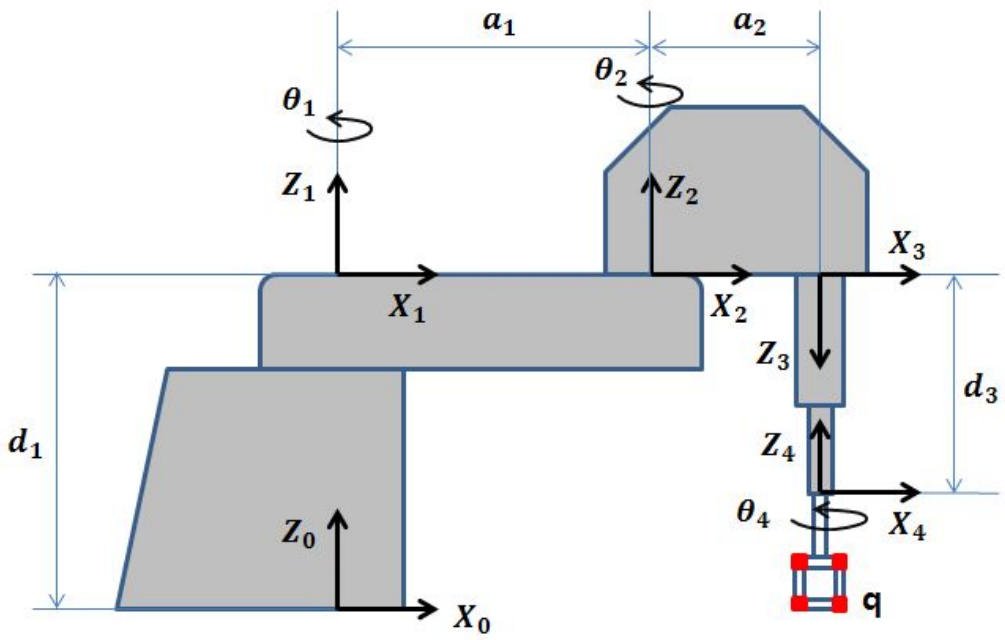


Fig. 2-1 Link parameters and link frame assignment of four axis robot

Table 2-1 Link parameters of four axis robot

axis	$\alpha_{i-1}(^\circ)$	$a_{i-1}(^\circ)$	$d_i(mm)$	$\theta_i(^\circ)$
1	0	0	387	θ_1
2	0	400	0	θ_2
3	180	250	d_3	0
4	-180	0	0	θ_4

Eq. (2-1) shows the general transform that defines frame $\{i\}$ relative to the frame $\{i+1\}$.

This transformation will be a function of four link parameters. For any robot, this transformation will be a function of only one variable while the remaining three parameters are fixed values.

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}d_i \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2\text{-}1) $$

Eq. (2-2) shows the each link transformation matrix ,T, calculated by applying the link parameters in Table 2-1 to Eq. (2-1).

$$ {}^{0}_{1}T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_1 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}_{3}T = \begin{bmatrix} 1 & 0 & 0 & a_2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{3}_{4}T = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ -\sin\theta_4 & -\cos\theta_4 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2\text{-}2) $$

The link transformations can be multiplied to find the single transformation that relates frame {4} to frame {0} as expressed in Eq. (2-3).

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

$$
{}^{0}_{4}T = \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_4) & -\sin(\theta_1+\theta_2+\theta_4) & 0 & a_2\cos(\theta_1+\theta_2)+a_1\cos\theta_1 \\ \sin(\theta_1+\theta_2+\theta_4) & \cos(\theta_1+\theta_2+\theta_4) & 0 & a_2\sin(\theta_1+\theta_2)+a_1\sin\theta_1 \\ 0 & 0 & 1 & -d_4+d_1-d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2-3}
$$

The position vector ,${}^{4}P^{j}$, of jth cue among the four cues attached to the robot end-effector in frame {4} is defined as Eq. (2-4).

$$
{}^{4}P^{j} = (P_x^j, P_y^j, P_z^j)^T \tag{2-4}
$$

The position vector ,F, of the cues with respect to the base frame {0} is expressed by Eq. (2-5).

$$
F = {}^{0}_{4}T\,{}^{4}P^{j} = \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_4) & -\sin(\theta_1+\theta_2+\theta_4) & 0 & a_2\cos(\theta_1+\theta_2)+a_1\cos\theta_1 \\ \sin(\theta_1+\theta_2+\theta_4) & \cos(\theta_1+\theta_2+\theta_4) & 0 & a_2\sin(\theta_1+\theta_2)+a_1\sin\theta_1 \\ 0 & 0 & 1 & -d_4+d_1-d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_x^j \\ P_y^j \\ P_z^j \\ 1 \end{bmatrix} \tag{2-5}
$$

Each position component of the robot forward kinematics in $x-y-z$ coordinate is shown in Eq. (2-6).

$$
\begin{aligned}
F_x^{i,j} &= \cos(\theta_1^i+\theta_2^i+\theta_4^i)P_x^j - \sin(\theta_1^i+\theta_2^i+\theta_4^i)P_y^j \\
&\quad + a_2\cos(\theta_1^i+\theta_2^i)+a_1\cos\theta_1^i \\
F_y^{i,j} &= \sin(\theta_1^i+\theta_2^i+\theta_4^i)P_x^j + \cos(\theta_1^i+\theta_2^i+\theta_4^i)P_y^j \\
&\quad + a_2\sin(\theta_1^i+\theta_2^i)+a_1\sin\theta_1^i \\
F_z^{i,j} &= P_z^j - d_4 - d_3^i + d_1
\end{aligned} \tag{2-6}
$$

where, i represents the number of moving point while the robot moves, $\theta_1^i, \theta_2^i, d_3^i$ and θ_4^i are the robot joint angles at the ith moving point of robot, and j represents the number

of cue. In addition, the link parameters ,a_1,a_2,d_1 and d_4, and the position vectors ,P_x^j,P_y^j and P_z^j, of the four cues of test model shown in Fig. 5-2 are expressed by Eq. (2-7).

$$
\begin{aligned}
&a_1 = 400\ mm,\ a_2 = 250\ mm,\ d_1 = 387\ mm,\ d_4 = 0\ mm \\
&P^1 = (P_x^1, P_y^1, P_z^1) = (15\ mm, -18\ mm, -103\ mm) \\
&P^2 = (P_x^2, P_y^2, P_z^2) = (15\ mm, -18\ mm, -133\ mm) \\
&P^3 = (P_x^3, P_y^3, P_z^3) = (-15\ mm, 18\ mm, -103\ mm) \\
&P^4 = (P_x^4, P_y^4, P_z^4) = (-15\ mm, 18\ mm, -133\ mm)
\end{aligned}
\tag{2-7}
$$

2.2 Robot inverse kinematics

Robot inverse kinematics is developed to calculate the joint angle of the robot for a specified target. If the given target position corresponded to F_d (X_d^j, Y_d^j and Z_d^j), then the corresponding joint angles ,$\theta_1^i, \theta_2^i, d_3^i$ and θ_4^i, of the robot are calculated by applying the Newton-Raphson method. The procedures are as follows.

step 1) In order to apply the Newton - Raphson method, the function ,f, is defined as Eq. (2-8).

$$f = {}_4^0T \cdot {}^4P - F_d \tag{2-8}$$

step 2) By using Eq. (2-6) and F_d, the function ,f, is obtained as follows.

$$\begin{bmatrix} f_x^j \\ f_y^j \\ f_z^j \end{bmatrix} = \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_4)P_x^j - \sin(\theta_1+\theta_2+\theta_4)P_y^j + a_2\cos(\theta_1+\theta_2) + a_1\cos\theta_1 - X_d^j \\ \sin(\theta_1+\theta_2+\theta_4)P_x^j + \cos(\theta_1+\theta_2+\theta_4)P_y^j + a_2\sin(\theta_1+\theta_2) + a_1\sin\theta_1 - Y_d^j \\ P_z^j - d_4 - d_3 + d_1 - Z_d^j \end{bmatrix} \tag{2-9}$$

step 3) The Newton-Raphson method of Eq. (2-10) is applied to the function ,f, given in Eq. (2-9).

$$x_{k+1} = x_k - \left[\frac{\partial f}{\partial x}(x_k)\right]^{-1} f(x_k) \tag{2-10}$$

where, $x = (\theta_1, \theta_2, d_3, \theta_4)^T$

$f = (f_x^j, f_y^j, f_z^j)^T$

step 4) Substituting Eq. (2-9) into Eq. (2-10), Eq. (2-10) is rewritten in the

matrix form.

$$\begin{bmatrix}\theta_{1,k+1}\\ \theta_{2,k+1}\\ d_{3,k+1}\\ \theta_{4,k+1}\end{bmatrix}=\begin{bmatrix}\theta_{1,k}\\ \theta_{2,k}\\ d_{3,k}\\ \theta_{4,k}\end{bmatrix}-\begin{bmatrix}\frac{\partial f_x^j}{\partial\theta_1} & \frac{\partial f_x^j}{\partial\theta_2} & \frac{\partial f_x^j}{\partial d_3} & \frac{\partial f_x^j}{\partial\theta_4}\\ \frac{\partial f_y^j}{\partial\theta_1} & \frac{\partial f_y^j}{\partial\theta_2} & \frac{\partial f_y^j}{\partial d_3} & \frac{\partial f_y^j}{\partial\theta_4}\\ \frac{\partial f_z^j}{\partial\theta_1} & \frac{\partial f_z^j}{\partial\theta_2} & \frac{\partial f_z^j}{\partial d_3} & \frac{\partial f_z^j}{\partial\theta_4}\end{bmatrix}^{-1}\cdot\begin{bmatrix}f_x^j(\theta_k)\\ f_y^j(\theta_k)\\ f_z^j(\theta_k)\end{bmatrix} \tag{2-11}$$

Eq. (2-11) can be expressed simply as Eq. (2-12).

$$\theta_{i,k+1}=\theta_{i,k}-\left[\ A\ \right]^{-1}\left[\ R\ \right] \tag{2-12}$$

step 5) In step 4, the inverse matrix is not a square matrix in general, so it can not be solved by a Gauss elimination method. Thus, when the size of the inverse matrix ,A, is $n\times m$, it is obtained by the method of Eq. (2-13).

i) In case of $n>m$; $A^{-1}\Rightarrow(A^TA)^{-1}A^T$ (2-13)

ii) In case of $n<m$; $A^{-1}\Rightarrow A^T(AA^T)^{-1}$

step 6) By combing Eqs. (2-12) and (2-13), Eq. (2-14) becomes.

$$\theta_{i,k+1}=\theta_{i,k}-(A^TA)^{-1}A^TR \tag{2-14}$$

Where, $\theta_{i,k}$ represents an arbitrarily given initial value. The matrix ,A, and the matrix ,R, are expressed by Eq. (2-15) and Eq. (2-16), respectively.

$$A=\begin{bmatrix} \frac{\partial f_x^1}{\partial \theta_1} & \frac{\partial f_x^1}{\partial \theta_2} & \frac{\partial f_x^1}{\partial d_3} & \frac{\partial f_x^1}{\partial \theta_4} \\ \frac{\partial f_y^1}{\partial \theta_1} & \frac{\partial f_y^1}{\partial \theta_2} & \frac{\partial f_y^1}{\partial d_3} & \frac{\partial f_y^1}{\partial \theta_4} \\ \frac{\partial f_z^1}{\partial \theta_1} & \frac{\partial f_z^1}{\partial \theta_2} & \frac{\partial f_z^1}{\partial d_3} & \frac{\partial f_z^1}{\partial \theta_4} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_x^4}{\partial \theta_1} & \frac{\partial f_x^4}{\partial \theta_2} & \frac{\partial f_x^4}{\partial d_3} & \frac{\partial f_x^4}{\partial \theta_4} \\ \frac{\partial f_y^4}{\partial \theta_1} & \frac{\partial f_y^4}{\partial \theta_2} & \frac{\partial f_y^4}{\partial d_3} & \frac{\partial f_y^4}{\partial \theta_4} \\ \frac{\partial f_z^4}{\partial \theta_1} & \frac{\partial f_z^4}{\partial \theta_2} & \frac{\partial f_z^4}{\partial d_3} & \frac{\partial f_z^4}{\partial \theta_4} \end{bmatrix}^{-1} \tag{2-15}$$

$$R=\begin{bmatrix} f_x^1(\theta_k) \\ f_y^1(\theta_k) \\ f_z^1(\theta_k) \\ \vdots \\ f_x^4(\theta_k) \\ f_y^4(\theta_k) \\ f_z^4(\theta_k) \end{bmatrix} \tag{2-16}$$

where,

$$\frac{\partial f_x^j}{\partial \theta_1}=-\sin(\theta_1+\theta_2+\theta_4)P_x^j-\cos(\theta_1+\theta_2+\theta_4)P_y^j-a_2\sin(\theta_1+\theta_2)-a_1\sin\theta_1$$

$$\frac{\partial f_x^j}{\partial \theta_2}=-\sin(\theta_1+\theta_2+\theta_4)P_x^j-\cos(\theta_1+\theta_2+\theta_4)P_y^j-a_2\sin(\theta_1+\theta_2)$$

$$\frac{\partial f_x^j}{\partial d_3}=0$$

$$\frac{\partial f_x^j}{\partial \theta_4}=-\sin(\theta_1+\theta_2+\theta_4)P_x^j-\cos(\theta_1+\theta_2+\theta_4)P_y^j$$

$$\frac{\partial f_y^j}{\partial \theta_1} = \cos(\theta_1+\theta_2+\theta_4)P_x^j - \sin(\theta_1+\theta_2+\theta_4)P_y^j + a_2\cos(\theta_1+\theta_2) + a_1\cos\theta_1$$

$$\frac{\partial f_y^j}{\partial \theta_2} = \cos(\theta_1+\theta_2+\theta_4)P_x^j - \sin(\theta_1+\theta_2+\theta_4)P_y^j + a_2\cos(\theta_1+\theta_2)$$

$$\frac{\partial f_x^j}{\partial d_3} = 0$$

$$\frac{\partial f_y^j}{\partial \theta_4} = \cos(\theta_1+\theta_2+\theta_4)P_x^j - \sin(\theta_1+\theta_2+\theta_4)P_y^j$$

$$\frac{\partial f_z^j}{\partial \theta_1} = 0$$

$$\frac{\partial f_y^j}{\partial \theta_2} = 0$$

$$\frac{\partial f_x^j}{\partial d_3} = -1$$

$$\frac{\partial f_y^j}{\partial \theta_4} = 0$$

step 7) Repeat step 6 until the given tolerance error (ε) is satisfied.

$$\sum_{i=1}^{4} \left|\theta_{i,k+1} - \theta_{i,k}\right| < \varepsilon \qquad (2\text{-}17)$$

2.3 Vision system model

The vision system model used in this study involves six camera parameters ,$C_1 \sim C_6$. It converts the position of an object in a 3-D space into the position of object in a 2-D camera plane. Among six camera parameters, $C_1 \sim C_4$ represent the uncertainty about the focal length and direction of the camera. C_5 and C_6 explain the uncertainty about the relative position between the camera and the robot. The vision system model is shown in Eq. (2-18).

$$\begin{bmatrix} X_m^{i,j} \\ Y_m^{i,j} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \end{bmatrix} \begin{bmatrix} F_x^{i,j} \\ F_y^{i,j} \\ F_z^{i,j} \end{bmatrix} + \begin{bmatrix} C_5 \\ C_6 \end{bmatrix} \tag{2-18}$$

where,

$$\begin{aligned} &C_{11} = C_1^2 + C_2^2 - C_3^2 - C_4^2, \quad C_{12} = 2(C_2C_3 + C_1C_4), \\ &C_{13} = 2(C_2C_4 - C_1C_3), \quad C_{21} = 2(C_2C_3 - C_1C_4), \\ &C_{22} = C_1^2 - C_2^2 + C_3^2 - C_4^2, \quad C_{23} = 2(C_3C_4 + C_1C_2) \end{aligned} \tag{2-19}$$

Substituting Eq. (2-19) into Eq. (2-18), Eq. (2-20) becomes.

$$\begin{aligned} X_m^{i,j} &= (C_1^2 + C_2^2 - C_3^2 - C_4^2)F_x^{i,j} + 2(C_2C_3 + C_1C_4)F_y^{i,j} \\ &\quad + 2(C_2C_4 - C_1C_3)F_z^{i,j} + C_5 \\ Y_m^{i,j} &= 2(C_2C_3 - C_1C_4)F_x^{i,j} + (C_1^2 - C_2^2 + C_3^2 - C_4^2)F_y^{i,j} \\ &\quad + 2(C_3C_4 + C_1C_2)F_z^{i,j} + C_6 \end{aligned} \tag{2-20}$$

Where, $F_x^{i,j}, F_y^{i,j}$ and $F_z^{i,j}$ represent the position of jth cue at the ith robot moving point in the 3-D space. $X_m^{i,j}$ and $Y_m^{i,j}$ represent the position of jth cue at the ith robot moving point in 2-D camera plane.

Chapter 3. Mathematical modeling for parameter estimation

3.1 N-R method

The robot vision control schemes proposed in the N-R method utilize a weighting matrix to weigh the obtained recent data near the target while the robot moves. Fig. 3-1 shows the overall flow divided into two stages. The first stage corresponds to the initial stage to obtain the optimal weighting factor. The second stage involves performing a rigid body placement task using the optimal weighting factor obtained in the initial stage. The mathematical modelings of the required camera parameters, robot joint angle estimation models, and weighting matrix model at each stage are discussed below.

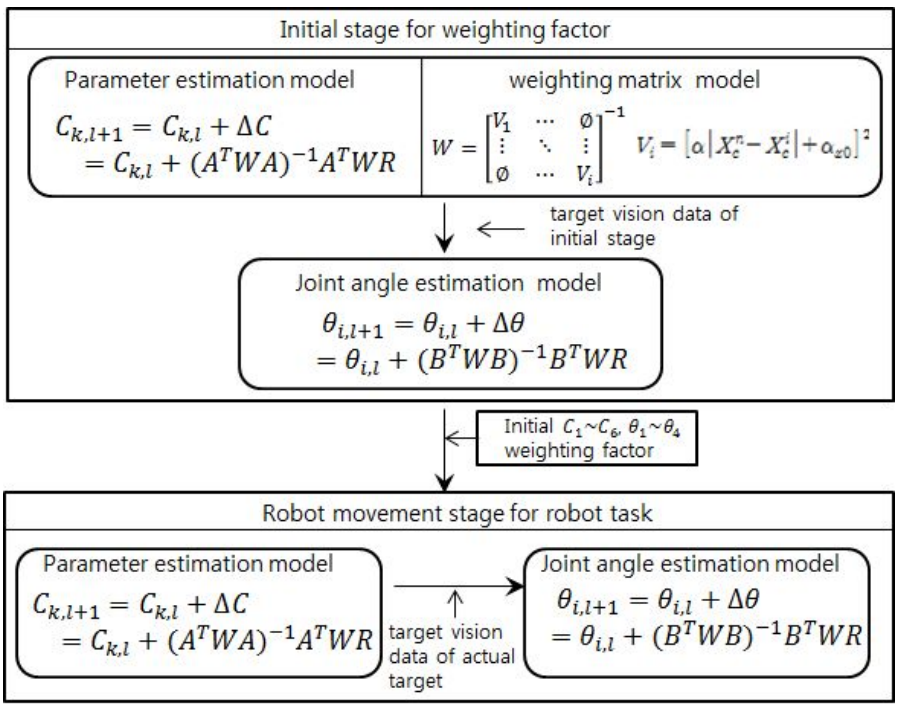


Fig. 3-1 Block diagram of robot's vision control scheme in N-R method

3.1.1 Camera parameter estimation model

This estimation model is concerned with determination of the unknown six camera parameters ,$C_1 \sim C_6$. These parameters included in the vision system model are used in order to transform the 3-D positions of cues into the 2-D camera image plane. For this study, vision data and robot joint angles at each moving point need to be obtained while robot moves toward a target along the arbitrary trajectory. Then, the performance index $J(C_k)$ is defined as Eq. (3-1)

$$J(C_k) = \sum_{i=1}^{n}\sum_{j=1}^{4}\left[X_m^{i,j}(C_k) - X_c^{i,j}\right]^2 + \left[Y_m^{i,j}(C_k) - Y_c^{i,j}\right]^2 \tag{3-1}$$

Where, k represents the number of parameters, and $X_c^{i,j}$ and $Y_c^{i,j}$ represent the actual vision data values of jth cue obtained through the camera at ith moving point of robot

By minimizing Eq. (3-1) using the N-R method, the relationship for camera parameter estimation is obtained as Eq. (3-2)

$$\begin{aligned} C_{k,l+1} &= C_{k,l} + \Delta C \\ &= C_{k,l} + (A^T W A)^{-1} A^T W R \end{aligned} \tag{3-2}$$

where, l represents the number of iteration in the calculation process. A is a jacobian matrix with a size of $(2\times i\times j)\times 6$, R is a residual vector with a size of $(2\times i\times j)\times 1$. Two matrices, A and R are given in Eqs. (3-3) and (3-4), respectively.

$$A=\begin{bmatrix} \frac{\partial X_m^{1,1}}{\partial C_1} & \frac{\partial X_m^{1,1}}{\partial C_2} & \frac{\partial X_m^{1,1}}{\partial C_3} & \frac{\partial X_m^{1,1}}{\partial C_4} & \frac{\partial X_m^{1,1}}{\partial C_5} & \frac{\partial X_m^{1,1}}{\partial C_6} \\ \frac{\partial Y_m^{1,1}}{\partial C_1} & \frac{\partial Y_m^{1,1}}{\partial C_2} & \frac{\partial Y_m^{1,1}}{\partial C_3} & \frac{\partial Y_m^{1,1}}{\partial C_4} & \frac{\partial Y_m^{1,1}}{\partial C_5} & \frac{\partial Y_m^{1,1}}{\partial C_6} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial Y_m^{i,j}}{\partial C_1} & \frac{\partial X_m^{i,j}}{\partial C_2} & \frac{\partial X_m^{i,j}}{\partial C_3} & \frac{\partial X_m^{i,j}}{\partial C_4} & \frac{\partial X_m^{i,j}}{\partial C_5} & \frac{\partial X_m^{i,j}}{\partial C_6} \\ \frac{\partial Y_m^{i,j}}{\partial C_1} & \frac{\partial Y_m^{i,j}}{\partial C_2} & \frac{\partial Y_m^{i,j}}{\partial C_3} & \frac{\partial Y_m^{i,j}}{\partial C_4} & \frac{\partial Y_m^{i,j}}{\partial C_5} & \frac{\partial Y_m^{i,j}}{\partial C_6} \end{bmatrix} \quad (3\text{-}3)$$

$$R=\begin{bmatrix} X_m^{1,1}-X_c^{1,1} \\ Y_m^{1,1}-Y_c^{1,1} \\ \vdots \\ X_m^{i,j}-X_c^{i,j} \\ Y_m^{i,j}-Y_c^{i,j} \end{bmatrix} \quad (3\text{-}4)$$

In particular, W represents a weighting matrix, and is an important factor in this study. The weight matrix model is developed in section 3.1.3. This estimation model is applied to each camera used for this study in order to calculate the six camera parameters.

3.1.2 Robot joint angle estimation model

The robot joint angle estimation model is concerned with determination of the robot joint angles ,$\theta_1 \sim \theta_4$, with respect to the specified target, based on the each camera parameters calculated in section 3.1.1. For this study, the performance index $J(\theta_k)$ is defined as Eq. (3-5).

$$J(\theta_k)=\sum_{q=1}^{3}\sum_{j=1}^{4}[X_m^{q,j}(F_x^j(\theta_i),F_y^j(\theta_i),F_z^j(\theta_i);C_k^q)-X_c^{q,j}]^2 + [Y_m^{q,j}(F_x^j(\theta_i),F_y^j(\theta_i),F_z^j(\theta_i);C_k^q)-Y_c^{q,j}]^2 \quad (3\text{-}5)$$

Where, q represents the number of cameras. $X_c^{q,j}$ and $Y_c^{q,j}$ represent the 2-D

camera space value of the jth cue in the qth camera for the target. $X_m^{q,j}$ and $Y_m^{q,j}$ represent the estimated vision system model values of the jth cue in the q th camera based on camera parameter ,$C_1 \sim C_6$.

By minimizing Eq. (3-5) using the N-R method, the relationship for joint angle estimation is obtained as Eq. (3-6).

$$\begin{aligned}\theta_{k,l+1} &= \theta_{k,l} + \Delta\theta \\ &= \theta_{k,l} + (B^T WB)^{-1} B^T WR\end{aligned} \tag{3-6}$$

Where, l represents the number of iteration in the calculation process. B is a jacobian matrix with a size of $(2\times q\times j)\times 4$, and is given by Eq. (3-7). R is a residual vector with a size of $(2\times q\times j)\times 1$, and is given by Eq. (3-8).

$$B=\begin{bmatrix} \frac{\partial X_m^{1,1}}{\partial\theta_1} & \frac{\partial X_m^{1,1}}{\partial\theta_2} & \frac{\partial X_m^{1,1}}{\partial\theta_3} & \frac{\partial X_m^{1,1}}{\partial\theta_4} \\ \frac{\partial Y_m^{1,1}}{\partial\theta_1} & \frac{\partial Y_m^{1,1}}{\partial\theta_2} & \frac{\partial Y_m^{1,1}}{\partial\theta_3} & \frac{\partial Y_m^{1,1}}{\partial\theta_4} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial X_m^{q,j}}{\partial\theta_1} & \frac{\partial X_m^{q,j}}{\partial\theta_2} & \frac{\partial X_m^{q,j}}{\partial\theta_3} & \frac{\partial X_m^{q,j}}{\partial\theta_4} \\ \frac{\partial Y_m^{q,j}}{\partial\theta_1} & \frac{\partial Y_m^{q,j}}{\partial\theta_2} & \frac{\partial Y_m^{q,j}}{\partial\theta_3} & \frac{\partial Y_m^{q,j}}{\partial\theta_4} \end{bmatrix} \tag{3-7}$$

$$R=\begin{bmatrix} X_m^{1,1}-X_c^{1,1} \\ Y_m^{1,1}-Y_c^{1,1} \\ \vdots \\ X_m^{q,j}-X_c^{q,j} \\ Y_m^{q,j}-Y_c^{q,j} \end{bmatrix} \tag{3-8}$$

In addition, weighting matrix ,W, in Eq. (3-6) uses a unit matrix to give equal weight to the three cameras used in the joint angle estimation model.

3.1.3 Weighting matrix model

In general, the weighting matrix ,W, of the parameter estimation model shown in Eq. (3-2) is used as a unit matrix to give an equal weight to all data obtained during robot movements. However, this study involves the application of the weighting matrix ,W, that provided a larger weight to the recent vision data obtained while the robot moves toward the target for precise robot vision control. The weighting matrix ,W, is defined as a diagonal matrix with a size of $(2\times i\times j)\times(2\times i\times j)$ as shown in Eqs. (3-9) and (3-10).

$$W=\begin{bmatrix} \ddots & & & & & & & & & & & \\ & \ddots & & & & & & & & & & \\ & & V_{8i-7}^{i,1} & & & & & & & & & \\ & & & V_{8i-6}^{i,1} & & & & \varnothing & & & & \\ & & & & V_{8i-5}^{i,2} & & & & & & & \\ & & & & & V_{8i-4}^{i,2} & & & & & & \\ & & & & & & V_{8i-3}^{i,3} & & & & & \\ & & & & & & & V_{8i-2}^{i,3} & & & & \\ & & & \varnothing & & & & & V_{8i-1}^{i,4} & & & \\ & & & & & & & & & V_{8i}^{i,4} & & \\ & & & & & & & & & & \ddots & \\ & & & & & & & & & & & \ddots \end{bmatrix}^{-1} \tag{3-9}$$

where,

$$\begin{aligned} V_{8i-7}^{i,1} &= [\alpha\left| X_c^{n,1}-X_c^{i,1}\right|+a_{x0}]^2 \\ V_{8i-6}^{i,1} &= [\alpha\left| Y_c^{n,1}-Y_c^{i,1}\right|+a_{y0}]^2 \\ V_{8i-5}^{i,2} &= [\alpha\left| X_c^{n,2}-X_c^{i,2}\right|+a_{x0}]^2 \\ V_{8i-4}^{i,2} &= [\alpha\left| Y_c^{n,2}-Y_c^{i,2}\right|+a_{y0}]^2 \\ V_{8i-3}^{i,3} &= [\alpha\left| X_c^{n,3}-X_c^{i,3}\right|+a_{x0}]^2 \\ V_{8i-2}^{i,3} &= [\alpha\left| Y_c^{n,3}-Y_c^{i,3}\right|+a_{y0}]^2 \\ V_{8i-1}^{i,4} &= [\alpha\left| X_c^{n,4}-X_c^{i,4}\right|+a_{x0}]^2 \\ V_{8i}^{i,4} &= [\alpha\left| Y_c^{n,4}-Y_c^{i,4}\right|+a_{y0}]^2 \end{aligned} \tag{3-10}$$

where, $X_c^{n,j}$ and $Y_c^{n,j}$ represent the vision data X and Y of the jth cue at the final nth moving point of the robot. $X_c^{i,j}$ and $Y_c^{i,j}$ represent the vision data X and Y of the jth cue at the ith moving point of the robot. Additionally, α is a weighting factor used to weight the recent data near the target.

In the Eq. (3-10), we set $a_{x0} = a_{y0} = 1$ to distinguish between the cases when the weighting matrix ,W, is applied and when it is not applied. Furthermore, in the case of $\alpha = 0$, the weighting matrix ,W, is used as the unit matrix to assign an equal weight to all vision data obtained while the robot is moving. This means that weighting matrix ,W, is not applied. In order to apply the weighting matrix ,W, that placed a weight on the recent vision data obtained near the target, a new weighting matrix ,W, is calculated by applying the optimal weighting factor estimated in the control scheme shown in section 4.3.

3.2 EKF method

Robot vision control using the EKF method estimates six camera parameters for each camera included in the vision system model. The joint angles of the robot are estimated using the estimated camera parameters. The overall flow is divided into two stages as shown in Fig. 3-2.

The first stage involves the initial stage. It is extremely important to effectively calculate the initial state variables x_k^- and the initial error covariance P_k^- to apply the EKF method for the estimation of camera parameters and the robot joint angles. For this study, the Monte-Carlo method is used to accurately calculate the initial value. The second stage is divided into a measurement model and a prediction model as the robot moving stage. In the measurement model, camera parameters and error covariance for each camera at the each moving point of robot are obtained. Next, based on the estimated camera parameters and its error covariance, the robot joint angle and its error covariance are estimated for the target. In the prediction model, the estimated

values through the measurement model are set as the initial values of the next moving point of the robot.

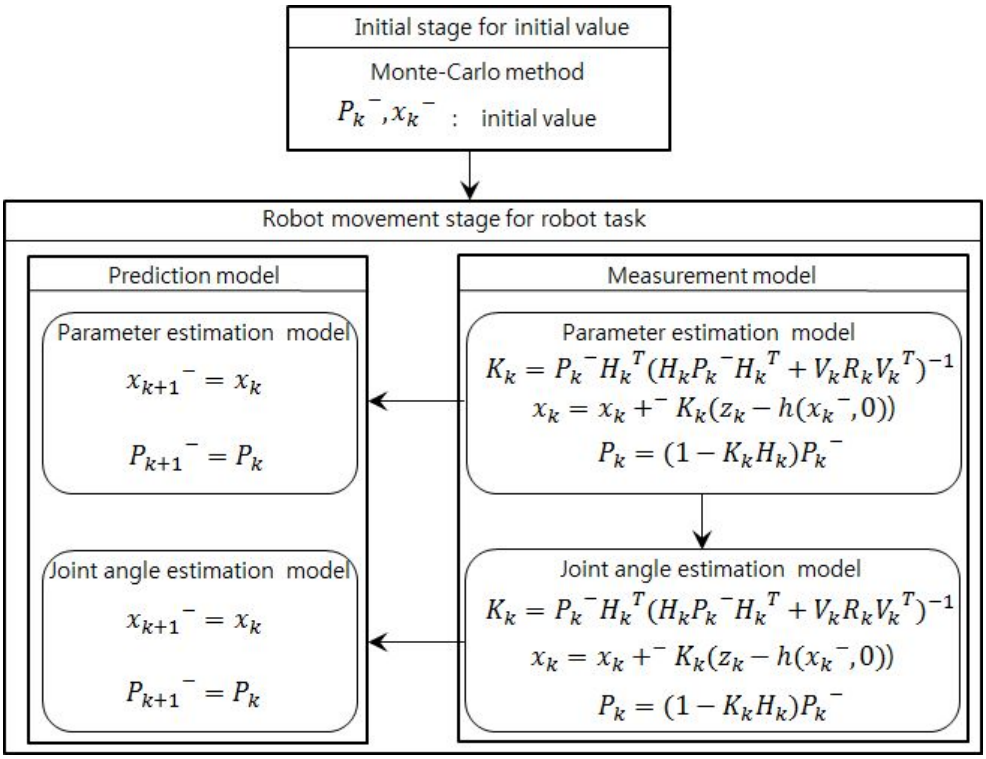


Fig. 3-2 Block diagram of robot's vision control scheme in EKF method

3.2.1 Initial value model

In order to apply the EKF method to the camera parameters and the robot joint angle estimation, it is very important to calculate the initial state variables and the initial error covariance effectively. As mentioned above, EKF method uses the Monte-Carlo method to calculate the initial values. The procedures for Monte-Carlo method are explained in detail in the control scheme on section 4.1.2.

3.2.2 Measurement model

(1) Camera parameter estimation

In order to estimate the uncertain six camera parameters of each camera included in the vision system model using the EKF method, it is necessary to define the nonlinear function ,h, shown in the parameter estimation model of Fig. 3-2.

The function ,h, for the measurement model is defined as Eq. (3-11) using the vision system model given in section 2.3, and has a size of $(2\times i\times j)\times 1$.

$$h\left(x_k^-,0\right)=\begin{bmatrix}h_x^{i,j}\\ h_y^{i,j}\end{bmatrix}=\begin{bmatrix}X_m^{i,j}\\ Y_m^{i,j}\end{bmatrix} \tag{3-11}$$

where,

$$\begin{aligned}X_m^{i,j}&=(C_1^2+C_2^2-C_3^2-C_4^2)F_x^{i,j}+2(C_2C_3+C_1C_4)F_y^{i,j}\\&\quad+2(C_2C_4-C_1C_3)F_z^{i,j}+C_5\\Y_m^{i,j}&=2(C_2C_3-C_1C_4)F_x^{i,j}+(C_1^2-C_2^2+C_3^2-C_4^2)F_y^{i,j}\\&\quad+2(C_3C_4+C_1C_2)F_z^{i,j}+C_6\end{aligned} \tag{3-12}$$

In addition, x_k^- is defined as Eq. (3-13).

$$x_k^-=\left[C_1\ C_3\ C_3\ C_4\ C_5\ C_6\right]^T \tag{3-13}$$

The equations of the measurement model shown in the parameter estimation model of Fig. 3-2 are composed of three components, the Kalman gain value,K_k, the camera parameter ,x_k, for each camera, and the camera error covariance P_k.

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \tag{3-14}$$

$$x_k = x_k^- + K_k (z_k - h(x_k^-, 0)) \tag{3-15}$$

$$P_k = (1 - K_k H_k) P_k^- \tag{3-16}$$

From Eq. (3-14), P_k^- with size of 6×6 represents an initial error covariance of camera parameter, which is used to calculate the Kalman gain ,K_k, with a size of 6×(2×i×j). Also, H_k is a Jacobian matrix with a size of (2×i×j)×6 as shown in Eq. (3-17). In addition, the influence component ,$V_k R_k V_k^T$, due to the noise of the measurement model is used as a unit matrix.

$$H_k = \frac{\partial h}{\partial x} = \frac{\partial h^{i,j}}{\partial C_{1\sim 6}} \tag{3-17}$$

$$= \begin{bmatrix} \frac{\partial h_x^{1,1}}{\partial C_1} & \frac{\partial h_x^{1,1}}{\partial C_2} & \frac{\partial h_x^{1,1}}{\partial C_3} & \frac{\partial h_x^{1,1}}{\partial C_4} & \frac{\partial h_x^{1,1}}{\partial C_5} & \frac{\partial h_x^{1,1}}{\partial C_6} \\ \frac{\partial h_y^{1,1}}{\partial C_1} & \frac{\partial h_y^{1,1}}{\partial C_2} & \frac{\partial h_y^{1,1}}{\partial C_3} & \frac{\partial h_y^{i,j}}{\partial C_4} & \frac{\partial h_y^{1,1}}{\partial C_5} & \frac{\partial h_y^{1,1}}{\partial C_6} \\ \frac{\partial h_x^{1,2}}{\partial C_1} & \frac{\partial h_x^{1,2}}{\partial C_2} & \frac{\partial h_x^{1,2}}{\partial C_3} & \frac{\partial h_x^{1,2}}{\partial C_4} & \frac{\partial h_x^{1,2}}{\partial C_5} & \frac{\partial h_x^{1,2}}{\partial C_6} \\ \frac{\partial h_y^{1,2}}{\partial C_1} & \frac{\partial h_y^{1,2}}{\partial C_2} & \frac{\partial h_y^{1,2}}{\partial C_3} & \frac{\partial h_y^{1,2}}{\partial C_4} & \frac{\partial h_y^{1,2}}{\partial C_5} & \frac{\partial h_y^{1,2}}{\partial C_6} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial h_x^{i,j}}{\partial C_1} & \frac{\partial h_x^{i,j}}{\partial C_2} & \frac{\partial h_x^{i,j}}{\partial C_3} & \frac{\partial h_x^{i,j}}{\partial C_4} & \frac{\partial h_x^{i,j}}{\partial C_5} & \frac{\partial h_x^{i,j}}{\partial C_6} \\ \frac{\partial h_y^{i,j}}{\partial C_1} & \frac{\partial h_y^{i,j}}{\partial C_2} & \frac{\partial h_y^{i,j}}{\partial C_3} & \frac{\partial h_y^{i,j}}{\partial C_4} & \frac{\partial h_y^{i,j}}{\partial C_5} & \frac{\partial h_y^{i,j}}{\partial C_6} \end{bmatrix}$$

In Eq. (3-15), z_k used for calculation of camera parameters ,x_k, is a matrix with the size of (2×i×j)×1 as shown in Eq. (3-18).

$$z_k^{i,j} = \left[X_c^{1,1}\ Y_c^{1,1}\ X_c^{1,2}\ Y_c^{1,2} \cdots X_c^{i,j}\ Y_c^{i,j} \right]^T \tag{3-18}$$

Where, $X_c^{i,j}$ and $Y_c^{i,j}$ represent the X and Y values of vision data measured from the camera to the jth cue at ith moving point while the robot is moving.

This camera parameter estimation model is applied to each camera used for this study.

(2) Robot joint angle estimation

The four joint angles ,$\theta_1 \sim \theta_4$, of the robot are estimated based on the estimated camera parameters ,$C_1 \sim C_6$.

In order to estimate the joint angles, the function ,h, which is described in the robot joint angle estimation model of Fig.3-2, is defined with the size of $(2\times q\times j)\times 1$ as shown in Eq. (3-19).

$$h(x_k,0)=\begin{bmatrix} h_x^{q,j} \\ h_y^{q,j} \end{bmatrix}=\begin{bmatrix} X_m^{q,j} \\ Y_m^{q,j} \end{bmatrix} \tag{3-19}$$

where,

$$\begin{aligned} X_m^{q,j} &= (C_1^2+C_2^2-C_3^2-C_4^2)^q F_x^j(\theta_i)+2(C_2C_3+C_1C_4)^q F_y^j(\theta_i) \\ &\quad +2(C_2C_4-C_1C_3)^q F_z^j(\theta_i)+C_5^q \\ Y_m^{q,j} &= 2(C_2C_3-C_1C_4)^q F_x^j(\theta_i)+(C_1^2-C_2^2+C_3^2-C_4^2)^q F_y^j(\theta_i) \\ &\quad +2(C_3C_4+C_1C_2)^q F_z^j(\theta_i)+C_6^q \end{aligned} \tag{3-20}$$

In Eq. (3-20), $q(1\sim 3)$ represents the number of cameras, and $X_m^{q,j}$ and $Y_m^{q,j}$ represent the X and Y values of the estimated vision system model to the jth cue of target in the qth camera while the robot is moving. Also, F_x^j, F_y^j andF_z^j represent the robot kinematic model value involving unknown robot joint angle ,θ_i. These values will be estimated as values of target. In addition, x_k^- is defined as Eq. (3-21).

$$x_k^- = [\theta_1 \;\; \theta_2 \;\; d_3 \;\; \theta_4]^T \tag{3-21}$$

The equations of the measurement model shown in the joint angle estimation model of Fig. 3-2 are composed of the Kalman gain value ,K_k, the joint angle parameter ,x_k, of robot, and the joint angle error covariance P_k.

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \tag{3-22}$$

$$x_k = x_k^- + K_k(z_k - h(x_k^-, 0)) \tag{3-23}$$

$$P_k = (1 - K_k H_k) P_k^- \tag{3-24}$$

The P_k^-, which needs to calculate the Kalman gain value K_k with a size of $4\times(2\times q\times j)$ in Eq. (3-22), represents an initial joint angle error covariance with a size of 4×4. H_k is a Jacobian matrix with a size of $(2\times q\times j)\times4$ as shown in Eq. (3-25). In addition, the influence component ,$V_k R_k V_k^T$, due to the noise of the measurement model is used as unit matrix.

$$H_k = \frac{\partial h}{\partial x} = \frac{\partial h^{q,j}}{\partial \theta_i} \tag{3-25}$$

$$= \begin{bmatrix} \frac{\partial h_x^{1,1}}{\partial \theta_1} & \frac{\partial h_x^{1,1}}{\partial \theta_2} & \frac{\partial h_x^{1,1}}{\partial d_3} & \frac{\partial h_x^{1,1}}{\partial \theta_4} \\ \frac{\partial h_y^{1,1}}{\partial \theta_1} & \frac{\partial h_y^{1,1}}{\partial \theta_2} & \frac{\partial h_y^{1,1}}{\partial d_3} & \frac{\partial h_y^{1,1}}{\partial \theta_4} \\ \frac{\partial h_x^{1,2}}{\partial \theta_1} & \frac{\partial h_x^{1,2}}{\partial \theta_2} & \frac{\partial h_x^{1,2}}{\partial d_3} & \frac{\partial h_x^{1,2}}{\partial \theta_4} \\ \frac{\partial h_y^{1,2}}{\partial \theta_1} & \frac{\partial h_y^{1,2}}{\partial \theta_2} & \frac{\partial h_y^{1,2}}{\partial d_3} & \frac{\partial h_y^{1,2}}{\partial \theta_4} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial h_x^{q,j}}{\partial \theta_1} & \frac{\partial h_x^{q,j}}{\partial \theta_2} & \frac{\partial h_x^{q,j}}{\partial d_3} & \frac{\partial h_x^{q,j}}{\partial \theta_4} \\ \frac{\partial h_y^{q,j}}{\partial \theta_1} & \frac{\partial h_y^{q,j}}{\partial \theta_2} & \frac{\partial h_y^{q,j}}{\partial d_3} & \frac{\partial h_y^{q,j}}{\partial \theta_4} \end{bmatrix}$$

From Eq. (3-23), z_k is a matrix with the size of $(2\times q\times j)\times1$ as shown in Eq. (3-26).

$$z_k^{q,j} = \left[X_c^{1,1}\ Y_c^{1,1}\ X_c^{1,2}\ Y_c^{1,2} \cdots X_c^{q,j}\ Y_c^{q,j} \right]^T \tag{3-26}$$

where, $X_c^{q,j}$ and $Y_c^{q,j}$ represent the X and Y values of the vision data measured from qth camera to the jth cue of target.

3.2.3 prediction model

(1) Camera parameter prediction

All of the influence components due to noise are used as the unit matrix in the equation of the prediction model. Thus, the camera parameter and its error covariance are defined as Eq. (3-27) in order to use the initial value of the next step.

$$x_{k+1}^- = x_k\ , \quad P_{k+1}^- = P_k \tag{3-27}$$

where,

$x_k^- = \left[C_1\ C_3\ C_3\ C_4\ C_5\ C_6 \right]^T$

(2) Robot joint angle prediction

Like the camera parameter prediction, all of the influence components due to noise are used as the unit matrix in the prediction model. Thus, the joint angle parameter and its error covariance are defined as Eq. (3-28). These values are used as the initial value of the next step.

$$x_{k+1}^- = x_k\ , \quad P_{k+1}^- = P_k \tag{3-28}$$

where,

$x_k^- = \left[\theta_1\ \theta_2\ d_3\ \theta_4 \right]^T$

Chapter 4. Robot's vision control scheme for fixed target estimation

4.1 N-R method

4.1.1 Data processing procedure

The vision data is obtained by dividing the initial stage and robot movement stage as shown in Fig. 4-1. The initial stage is used to calculate the optimal weighting factor, and the case without the weighting matrix is not applied. The robot movement stage is used in the study to estimate the fixed target.

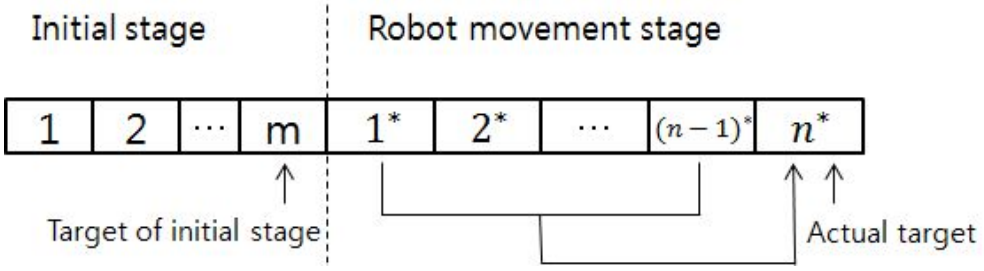


Fig. 4-1 Data processing procedures in N-R method

4.1.2 Control scheme without the weighting matrix

If the weighting matrix is not applied as in the previous study, the position of the fixed target is estimated by using the weighting matrix ,W, as a unit matrix in Eq. (3-2). The flow of the control scheme is shown in Fig. 4-2, and the explanation for each step is as follows.

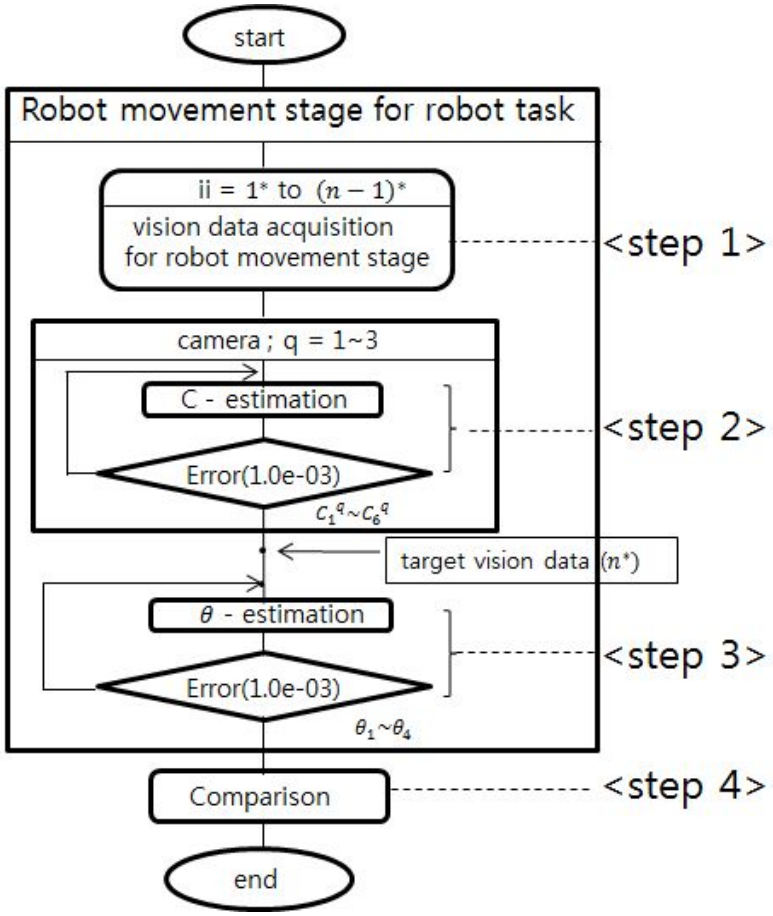


Fig. 4-2 Robot's vision control scheme of N-R method without weighting matrix

① step 1

Vision data for each of the four cues is acquired at each moving point by using the three cameras while moving towards a fixed target along the robot motion trajectory ranging from 1^* to $(n-1)^*$ except target.

② step 2

Vision data obtained in step 1 and any initial values are applied to the parameter estimation model to estimate the parameters for each camera.

③ step 3

Estimated parameters for each camera and vision data for target are applied to

the robot joint angle estimation model to estimate the robot joint angle with respect to the target.

④ step 4

The error value is calculated to compare the estimated joint angle with the actual joint angle.

4.1.3 Control scheme with weighting matrix

In the initial stage shown in Fig. 4-1, the optimal weighting factor is calculated by repeating the weighting factor with increment of 0.01, ranging from 0 to 0.5. The position value of the fixed target is calculated by applying the optimal weighting factor to vision data obtained in robot each moving point ranging from 1^{*} to $(n-1)^{*}$ except vision data of target

The flow of the control scheme is shown in Fig. 4-3, and the explanation for each step is as follows.

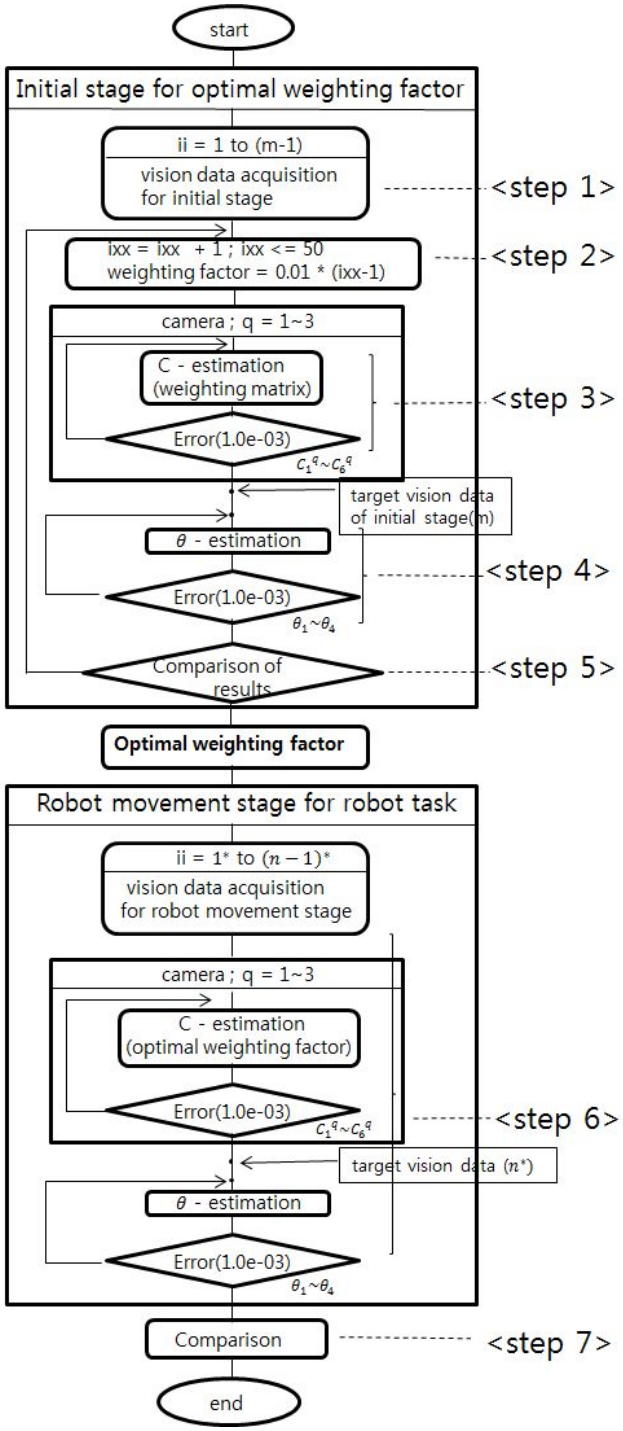


Fig. 4-3 Robot's vision control scheme of N-R method with weighting matrix

① step 1

In the initial stage of Fig. 4-1, vision data for each of the four cues is acquired while moving towards a fixed target along the motion trajectory ,ranging from 1 to $(n-1)$ except target, by using three cameras at each moving point.

② step 2

In order to find the optimal weighting factor at initial stages, the weighting factor is increased from 0 to 0.5, and it is repeated until the optimal weighting factor is calculated.

③ step 3

In the initial stage, the vision data obtained from each camera is applied to the parameter estimation model. The parameters for the robot movement stage are estimated using the weighting factor of step 2.

④ step 4

The estimated camera parameters, the vision data of the target point ,m, of the initial stage are applied to the joint angle estimation model to estimate the robot joint angle for initial target.

⑤ step 5

Calculation is continuously performed if the target position value in the initial stage estimated from the robot joint angle calculated at the most recent step is lower than the position value calculated at the previous step. However, if it becomes large, then the calculation is stopped, and the weighting factor used in the previous step is selected as the optimal weighting factor.

⑥ step 6

The positional value for the target is estimated by applying the optimal weighting factor calculated in step 5. The procedure is the same as method of the control scheme without the weighting matrix in section 4.1.2.

⑦ step 7

The error values are calculated to compare the estimated joint angle with the actual joint angle.

4.2 EKF method

4.2.1 Data processing procedure

The EKF method estimates the fixed target using the initial values calculated in the initial stage of the robot and the vision data of the first point ,1^*, of robot movement stage as shown in Fig. 4-4. The next step estimates the fixed target using the second point ,2^*, and the estimated values of the previous point as the initial value. The above process is then repeated until the last data point ,$(n-1)^*$, before the target.

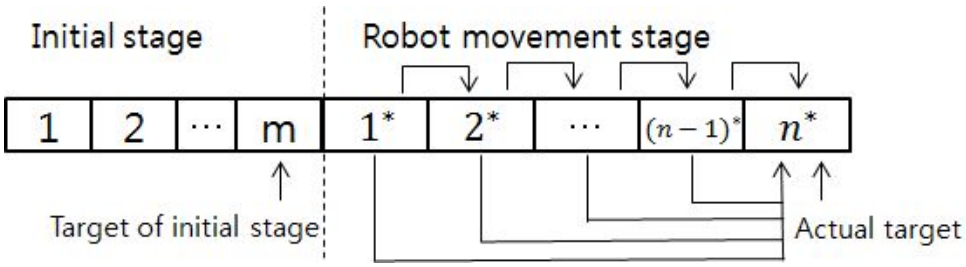


Fig. 4-4 Data processing procedures in EKF method

4.2.2 Control scheme of EKF method

Fig. 4-5 shows the flow of the control scheme of fixed target estimation using the EKF method.

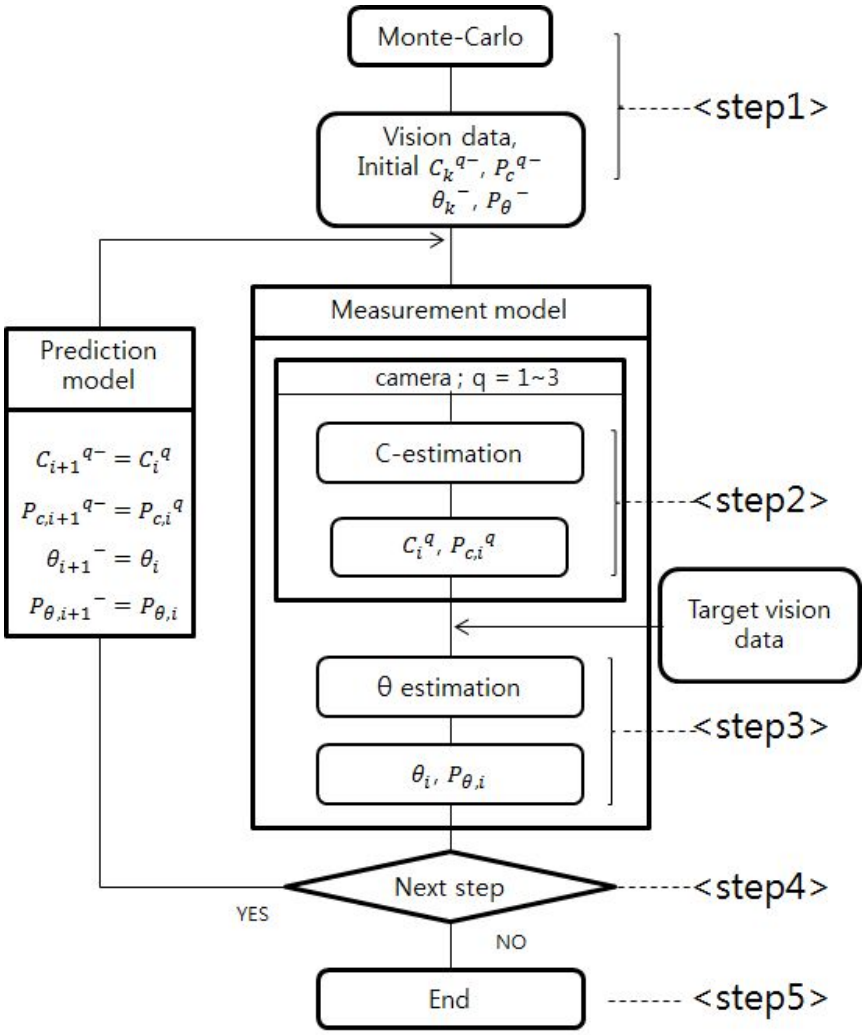


Fig. 4-5 Robot's vision control scheme of EKF method

① step 1

In the initial stage, initial values and error covariance for each camera parameter and initial values and error covariance for the joint angles are obtained through the Monte-Carlo procedure shown in Figs. 4-6. A detailed explanation is provided in Appendix A.

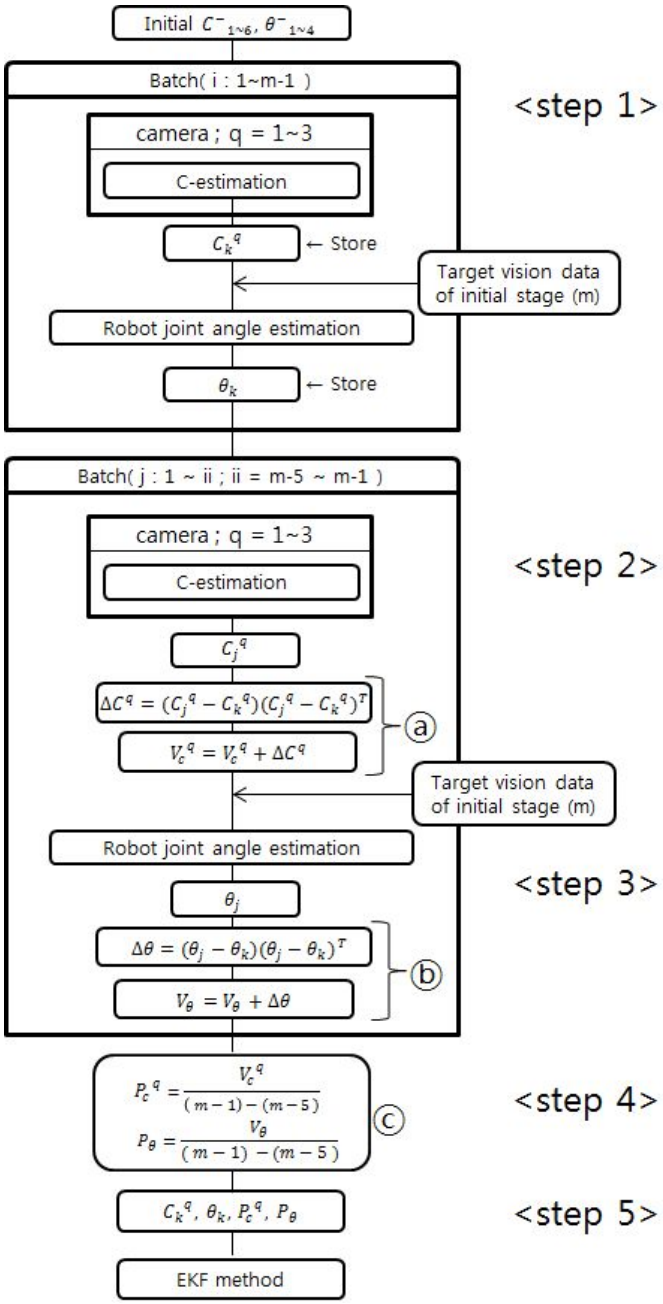


Fig. 4-6 Procedures of Monte-Carlo method

② step 2

The obtained initial values are substituted into the parameter estimation model by applying the EKF method. Hence, estimated values for each camera parameter and error covariance for the first moving point of the robot are obtained.

③ step 3

The estimated camera parameters and target vision data ,n^*, are substituted into the robot joint angle estimation model to estimate the robot joint angle and error covariance for the fixed target.

④ step 4

The parameters and error covariance for each camera estimated in step 2 and the robot joint angle and error covariance for the fixed target estimated in step 3 are used as initial values for the next point of robot movement through the predictive model.

⑤ step 5

step 2 ~ step 4 are repeated to estimate the position of the fixed target until the final robot moving point, $(n-1)^*$, and the error is shown by comparing the estimated position with the actual position value.

Chapter 5. Experimental equipment and experimental method

5.1 Experimental equipment composition

The experimental set-up used in this study consists of a PC system, a robot system, and a vision system as shown in Photo. 5-1 and Fig. 5-1. The PC system consisted of an Intel Pentium 4 processor, a 2.8 GHz CPU, and 512 MB RAM. Additionally, the robot system consisted of a Samsung SM7 four-axis scara robot and a Samsung CSD3 series servo drive robot controller. The vision system consists of a MATROX black-white meteor2-MC4 with a 640 x 480 resolution, a 256-image-level vision board, and a Sony XC ES51 CCD camera.

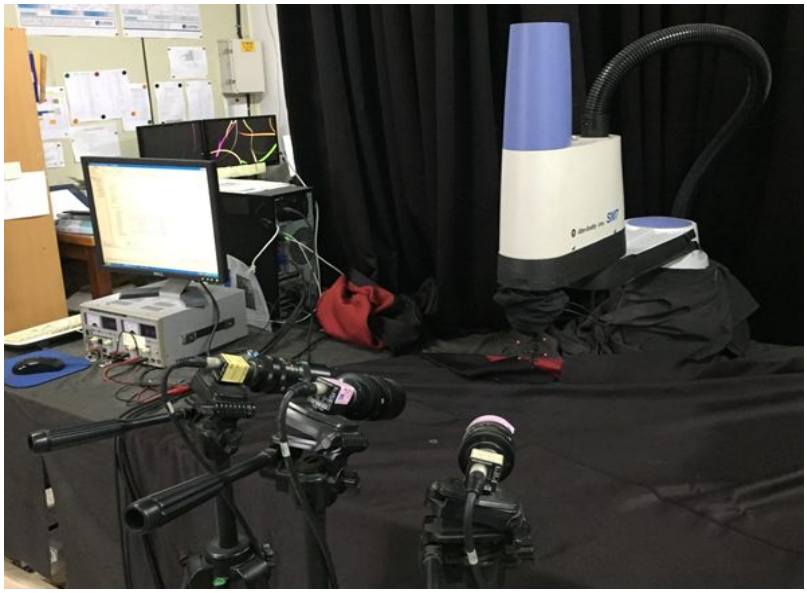


Photo. 5-1 Experimental set-up

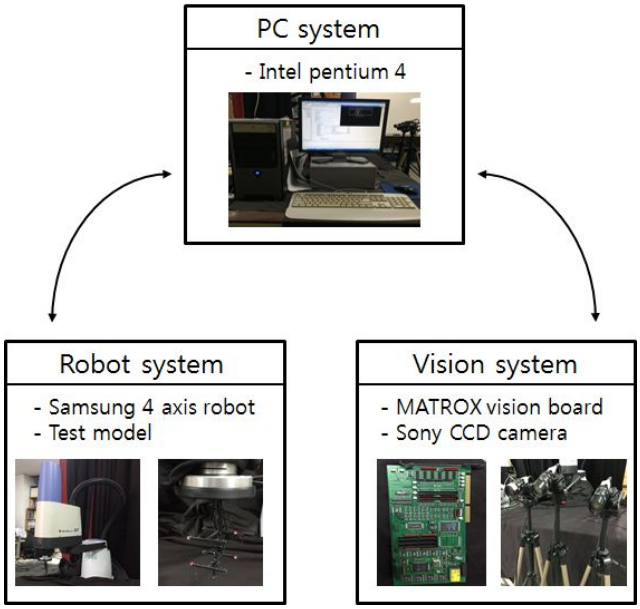


Fig. 5-1 Experimental schematic diagram

5.2 Test model

As shown in Fig.5-2, test model is manufactured to form various objects by using nine cues without using the whole object shape in order to improve the processing speed by minimizing the amount of vision data. In this study, the experiments are performed using four cues (cues 1, 2, 7, and 8) to form a rigid body.

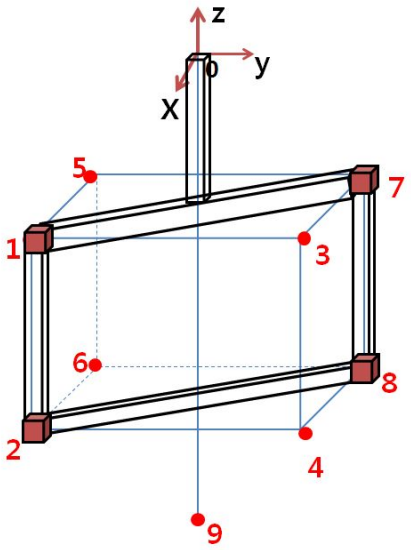


Fig. 5-2 Test model used for experiment

5.3 Experimental method

In the study, it is necessary to set the start point and the end point of the motion trajectory of the Samsung SM7 robot for fixed target estimation through the experiments of the real time rigid body placement task. Thus, the joint angles of the end point are obtained by using the joint angle values of target calculated from the inverse kinematic model of section 2.2. The joint angles of the starting point and the end point shown in Table 5-1 are calculated according to Eq. (5-1).

Table 5-1 Robot's joint angles of start point and end point for robot's trajectory

joint angles	start point	end point
θ_1 (degree)	-2.9822	5.5177
θ_2 (degree)	1.0822	3.5822
d_3 (mm)	105	124
θ_4 (degree)	-0.6011	-9.1011

Trajectory is set to forty points in which ten points and thirty points are allocated to the initial stage and robot movement stage, respectively as shown in Fig.5-3. In Eq. (5-1), $\theta_{k,s}(k=1\sim4)$ denotes each joint angle of the start point, and $\theta_{k,e}(k=1\sim4)$ denotes each joint angle of the end point.

$$\theta_{1,i}=\theta_{1,s}+\frac{(\theta_{1,e}-\theta_{1,s})}{40}(i-1) \qquad \theta_{2,i}=\theta_{2,s}+\frac{(\theta_{2,e}-\theta_{2,s})}{40}(i-1) \qquad (5\text{-}1)$$

$$d_{3,i}=d_{3,s}+\frac{(d_{3,e}-d_{3,s})}{40}(i-1) \qquad \theta_{4,i}=\theta_{4,s}+\frac{(\theta_{4,e}-\theta_{4,s})}{40}(i-1)$$

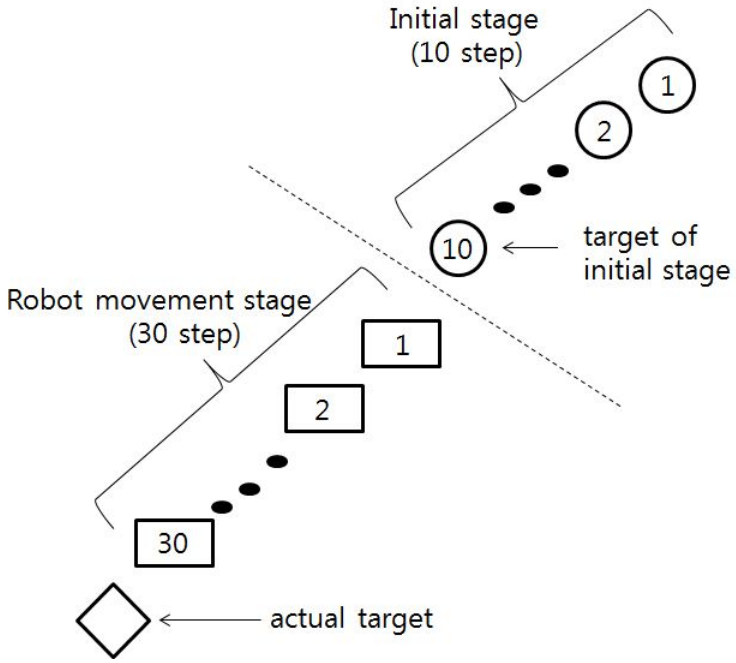


Fig. 5-3 Robot's trajectory

Chapter 6. Results of vision system model's suitability

6.1 Optimal weighting factor in N-R method

With respect to the N-R method, an initial stage with ten moving points of the robot is set to perform fixed target estimation for a real-time rigid placement task using the presented weighting matrix model in section 3.1.3. The optimal weighting factor ,α, is calculated for camera parameter estimation with high precision by applying the obtained vision data in the initial stage to the control method (step1 ~ step5) of section 4.1.3. As shown in Fig. 6-1 and listed in Table 6-1, the error value between the estimated position value the actual position value according to the increase of the weighting factor is greatest when the weighting factor is 0. Then error values decreases until the weighting factor is 0.17, and increases after weighting factor of 0.17. Thus, 0.17 is selected as an optimal weighting factor, and applied to Eqs. (3-9) and (3-10) for a new weighting matrix ,W. This weighting matrix is used to estimate the target value in the robot movement stage in section 6.2.2.

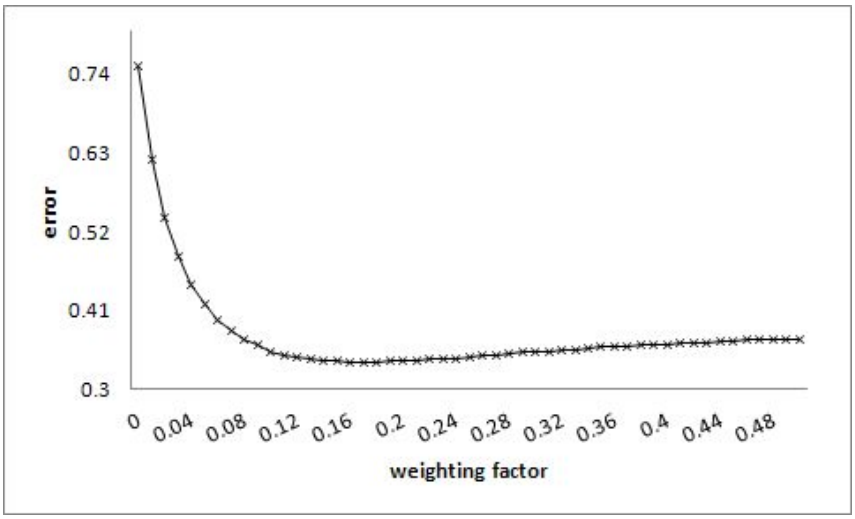


Fig. 6-1 Optimal weighting factor of N-R method in initial stage

Table 6-1 Errors according to variation of weighting factors (α) of N-R method in initial stage

α	error	α	error	α	error	α	error	α	error
0.01	0.6212	0.11	0.3479	0.21	0.3408	0.31	0.3537	0.41	0.3646
0.02	0.5387	0.12	0.3453	0.22	0.3421	0.32	0.3550	0.42	0.3654
0.03	0.4844	0.13	0.3426	0.23	0.3428	0.33	0.3560	0.43	0.3660
0.04	0.4464	0.14	0.3407	0.24	0.3435	0.34	0.3573	0.44	0.3671
0.05	0.4185	0.15	0.3392	0.25	0.3456	0.35	0.3588	0.45	0.3680
0.06	0.3980	0.16	0.3388	0.26	0.3469	0.36	0.3600	0.46	0.3687
0.07	0.3825	0.17	0.3386	0.27	0.3480	0.37	0.3606	0.47	0.3693
0.08	0.3706	0.18	0.3387	0.28	0.3492	0.38	0.3621	0.48	0.3699
0.09	0.3615	0.19	0.3392	0.29	0.3513	0.39	0.3626	0.49	0.3706
0.1	0.3535	0.2	0.3397	0.3	0.3521	0.4	0.3636	0.5	0.3703

6.2 Vision system model's suitability of N-R method

In order to estimate the fixed target value using the N-R method, first of all, the suitability of vision system model must be verified by comparing the vision system model values using the estimated camera parameters with the actual vision data acquired in each camera while the robot moves toward the fixed target. The r.m.s.[27] error between the actual vision data and the estimated value of the vision system model at the ith moving point in each camera is defined as shown in Eq. (6-1). Also, average error, $e_{avg,}$ is defined as Eq. (6-2).

$$e_{r.m.s}^{i} = \sqrt{\frac{\sum_{j=1}^{j^*}\left\{(e_x^{i,j})^2 + (e_y^{i,j})^2\right\}}{2 \times j^*}} \tag{6-1}$$

$$e_{avg} = \frac{\sum_{i=1}^{n} e_{r.m.s}^{i}}{n} \tag{6-2}$$

where, $e_x^{i,j}$ and $e_y^{i,j}$ represent the x and y error values of jth cue at ith moving point, respectively, and j^* represents the number of cue.

6.2.1 Without the weighting matrix

The vision system model value is calculated according to the control scheme of section 4.1.2 using the weighting matrix ,W, as the unit matrix when the weighting matrix is not applied. Especially, in order to compare the suitability of vision system model with the case of using the weighting matrix, the range of data used for the camera parameter estimation is classified into two groups; one is all data, and the order is ten data near target while the robot moves toward target.

(1) All data of robot movement stage

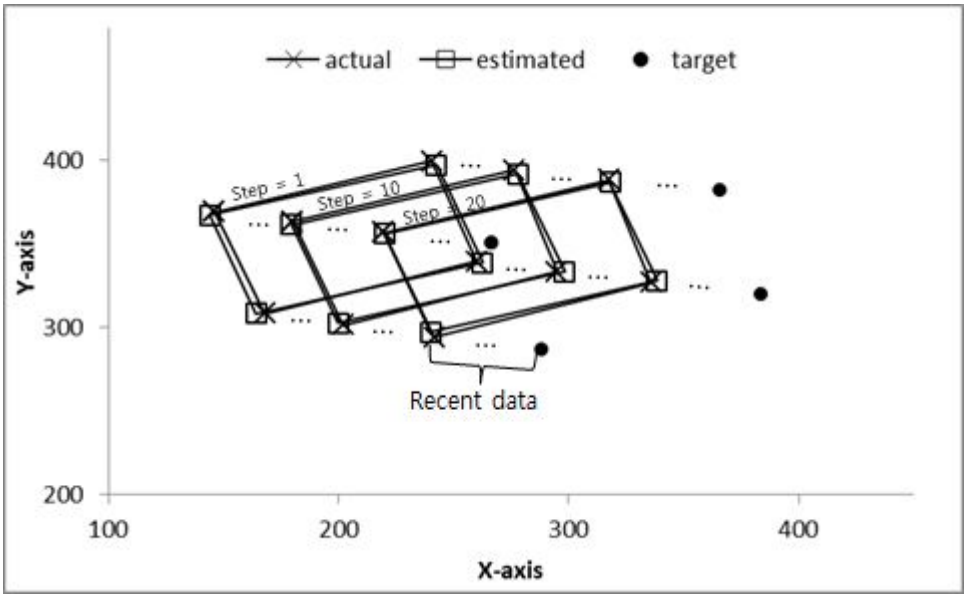
Table 6-2 shows the parameters for each camera without the weighting matrix by using all data of robot movement.

Table 6-2 The estimated six parameters using N-R method without weighting matrix

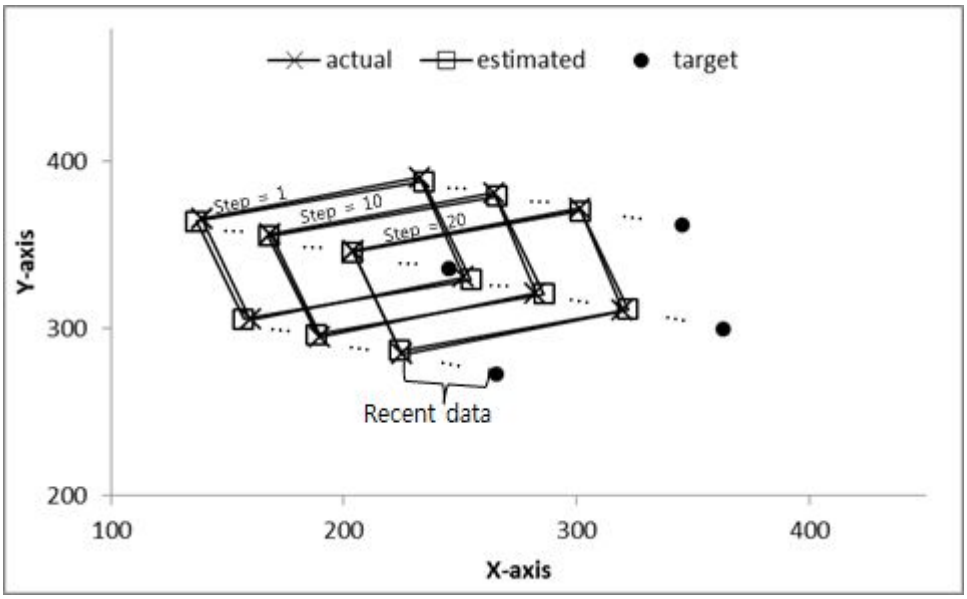
	Camera 1	Camera 2	Camera 3
C1	0.6543	0.5699	0.4831
C2	0.1984	0.1274	0.0963
C3	0.8288	0.8397	0.8862
C4	1.0262	1.0815	1.1313
C5	112.97778	1278.7608	1509.1439
C6	704.978	704.1501	593.5224

The results of comparing the estimated vision system model values by applying the parameters shown in Table 6-2 to Eq. (2-20) and the actual vision

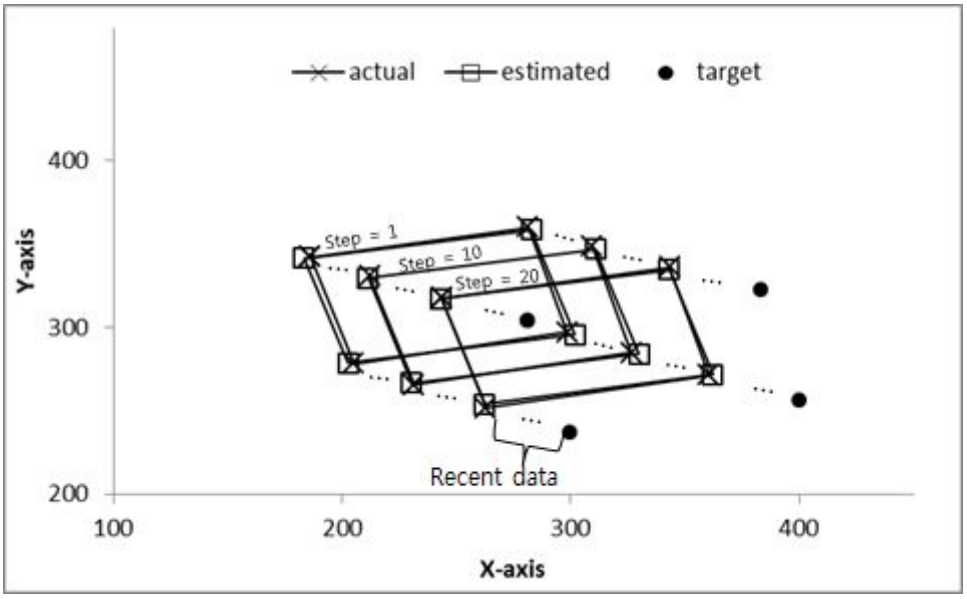
data obtained by the camera are shown in Fig. 6-2. Fig. 6-3 shows the calculated r.m.s. error values at each point of the robot movement by using Eq. (6-1). Also, the table 6-3 shows that the calculated average error values by using Eq. (6-2) in each camera correspond to 1.8844 *pixel* in camera 1, 1.7189 *pixel* in camera 2, and 1.5807 *pixel* in camera 3. Thus, it can be seen that it is suitability.



(a) camera 1



(b) camera 2



(c) camera 3

Fig. 6-2 Comparison of the actual vision data and estimated vision system model in N-R method without weighting matrix

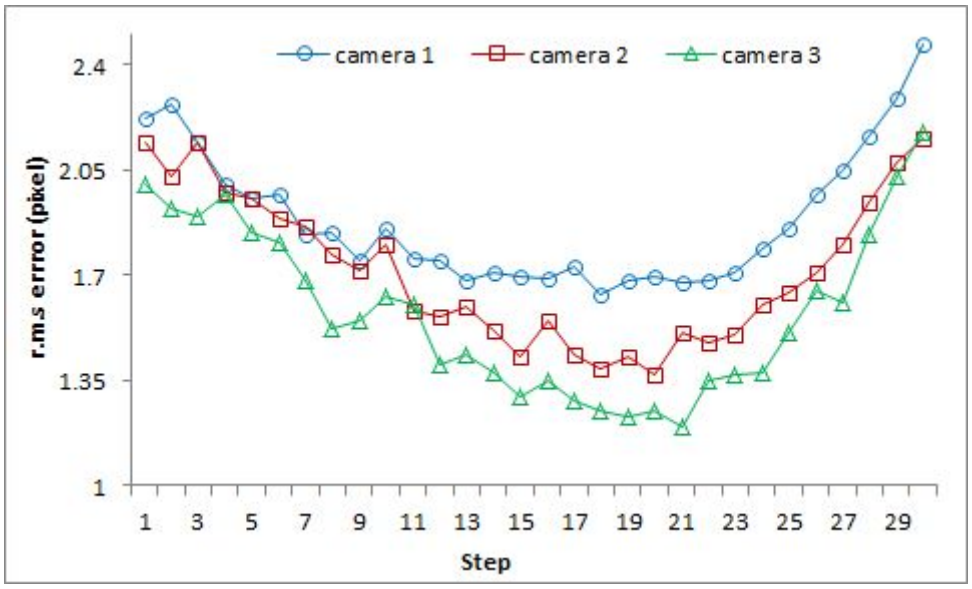


Fig. 6-3 The r.m.s. errors between actual vision data and estimated vision system model in N-R method without weighting matrix

Table 6-3 Average errors of each camera between actual vision data and estimated vision system model in N-R method without weighting matrix (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
Average error	1.8844	1.7189	1.5807

(2) Recent data of robot movement stage

The calculated r.m.s. error for each point by using Eq. (6-1) with ten data near the target in Fig. 6-2 are shown in Fig. 6-4. Also, the table 6-3 shows that the calculated average error values by using Eq. (6-3) in each camera correspond to 1.9646 *pixel* in camera 1, 1.7424 *pixel* in camera 2, and 1.6112 *pixel* in camera 3.

$$e_{avg}^{i} = \frac{\sum_{i=n-9}^{n} e_{r.m.s}^{i}}{10} \tag{6-3}$$

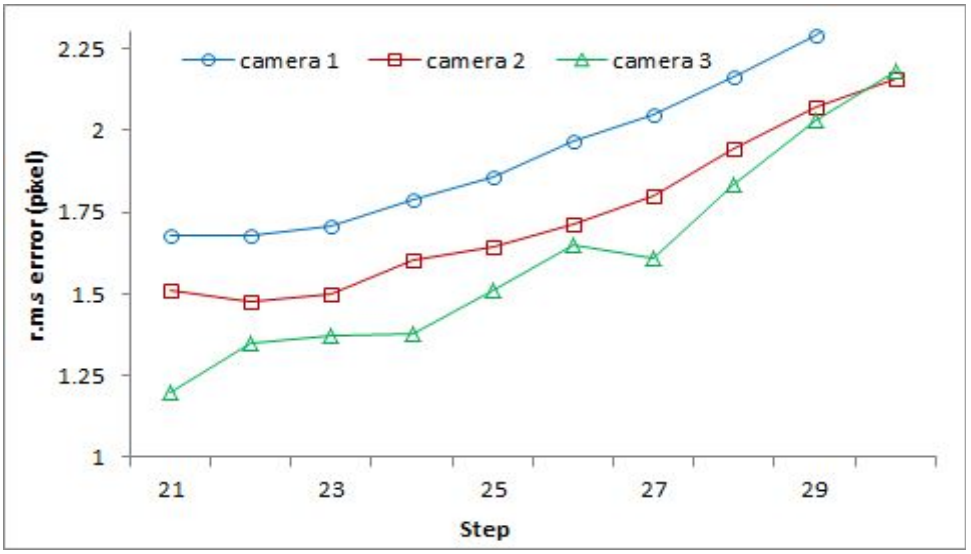


Fig. 6-4 The r.m.s. errors between actual vision data and estimated vision system model using the recent ten data in N-R method without weighting matrix

Table 6-4 Average errors of each camera between actual vision data and estimated vision system model using the recent ten data in N-R method without weighting matrix (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
Average error	1.9646	1.7424	1.6112

From Table 6-3 and Table 6-4, the results of both cases are approximately similar. Thus, It can be seen that the vision system model is suitable in both cases.

6.2.2 With the weighting matrix

The optimal weighting factor of 0.17 calculated in section 6.2 is used in the control scheme of section 4.1.3, in order to estimate parameters for the three cameras in the robot movement stage.

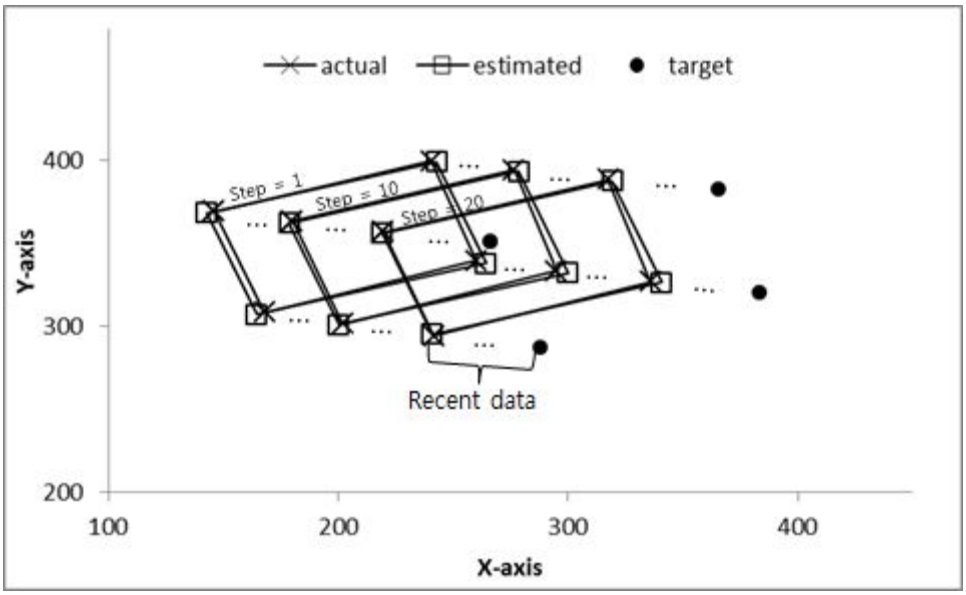
(1) All data of robot movement stage

Table 6-5 lists the parameters for each camera applying the weighting matrix by using the all data of the robot movement stage.

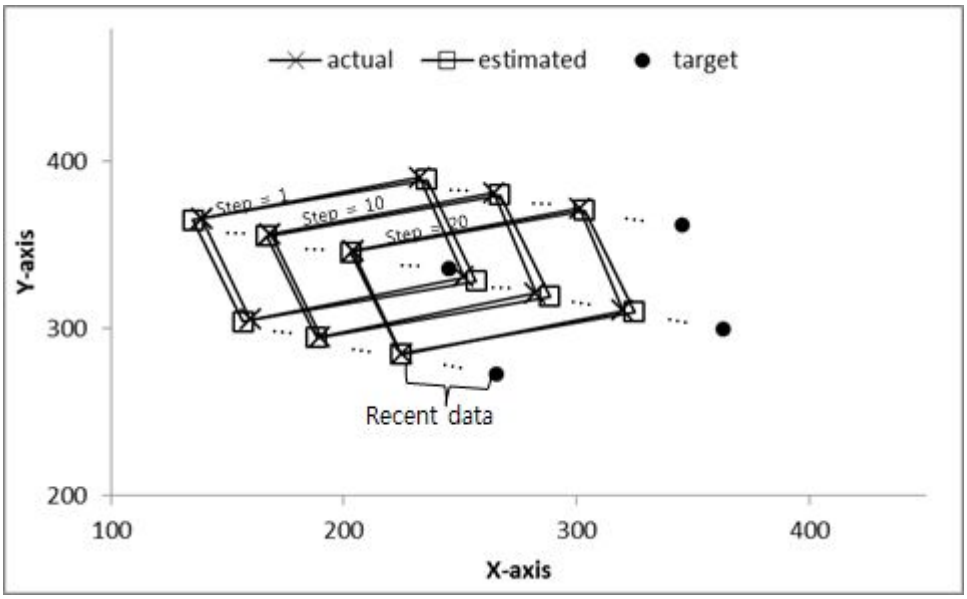
Table 6-5 The estimated six parameters using N-R method with weighting matrix

	Camera 1	Camera 2	Camera 3
C1	0.6454	0.5644	0.4779
C2	0.1840	0.1184	0.0914
C3	0.8576	0.8633	0.9081
C4	1.0512	1.0990	1.1446
C5	1196.2575	1340.8391	1562.2875
C6	710.5358	704.4669	585.8569

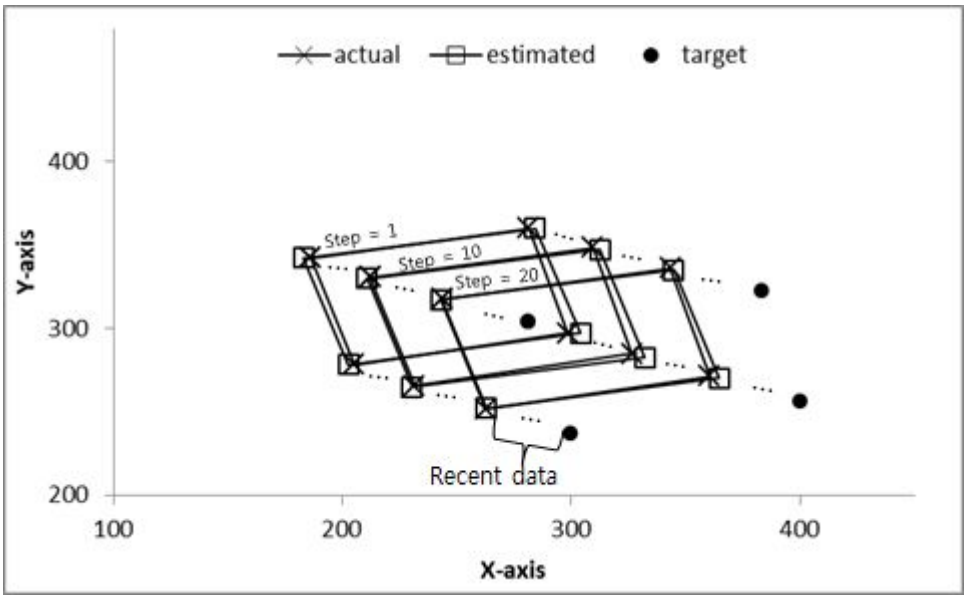
Fig. 6-5 shows the results of comparing the estimated vision system model values by applying the parameters listed in Table 6-5 to Eq. (2-20) and the actual vision data acquired by the camera. Fig. 6-6 shows the calculated r.m.s. error values at each point of the robot movement stage by applying Eq. (6-1). Additionally, as listed in Table 6-6, the calculated average error values in each camera by applying Eq. (6-2) correspond to 1.9302 *pixel* in camera 1, 2.1966 *pixel* in camera 2, and 2.0389 *pixel* in camera 3. Thus, this exceeds the values without the weighting matrix as listed in Table 6-3.



(a) camera 1



(b) camera 2



(c) camera 3

Fig. 6-5 Comparison of the actual vision data and estimated vision system model in N-R method with weighting matrix

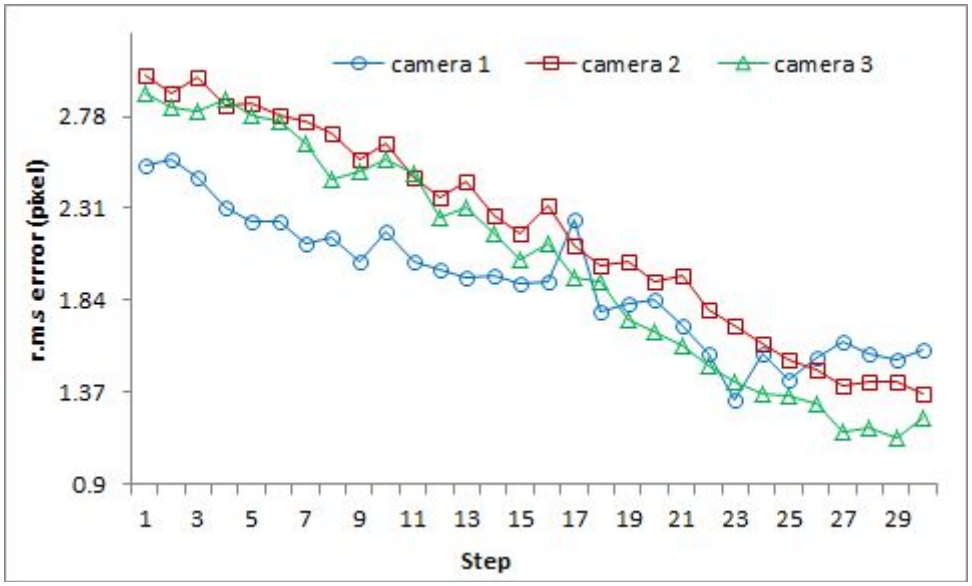


Fig. 6-6 The r.m.s. errors between actual vision data and estimated vision system model in N-R method with weighting matrix

Table 6-6 Average errors of each camera between actual vision data and estimated vision system model in N-R method with weighting matrix (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
Average error	1.9302	2.1966	2.0389

(2) Recent data of robot movement stage

Fig. 6-7 shows the calculated r.m.s error for each point by applying Eq. (6-1) with ten data points near the target in Fig. 6-5. Table 6-7 lists that the average error values calculated by Eq. (6-3) in each camera correspond to 1.5492 *pixel* in camera 1, 1.5749 *pixel* in camera 2, and 1.3322 *pixel* in camera 3. Thus, using the recent ten data points leads to lower calculated average error values when compared to those without the weighting matrix as listed in Table 6-4.

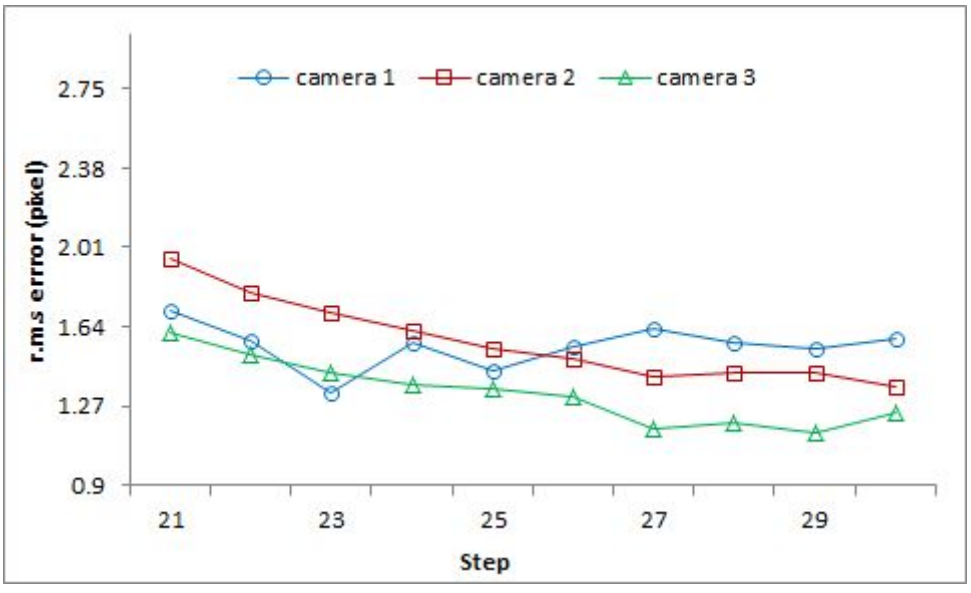


Fig. 6-7 The r.m.s. errors between actual vision data and estimated vision system model using the recent ten data in N-R method with weighting matrix

Table 6-7 Average errors of each camera between actual vision data and estimated vision system model using the recent ten data in N-R method with weighting matrix (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
Average error	1.5492	1.5749	1.3322

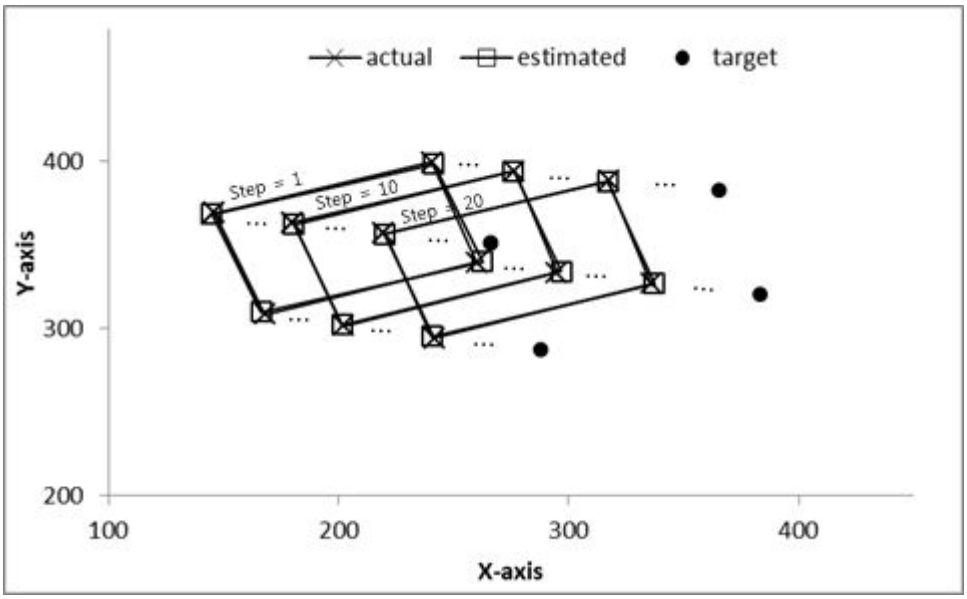
6.3 Vision system model's suitability of EKF method

The initial value setting is very important when using the EKF method. Thus, it is necessary to set the initial stage to determine the initial values and the error covariance for the camera parameters and the robot joint angle. The initial values and error covariance values in calculated the initial stage are used to show suitability of the vision system model for the fixed target estimation using the EKF method shown in section 4.2. Vision system model values using estimated camera parameters are compared with actual vision data at each point acquired by each camera during robot movement. Table 6-8 lists the estimated parameters for the three cameras at each point of robot movement.

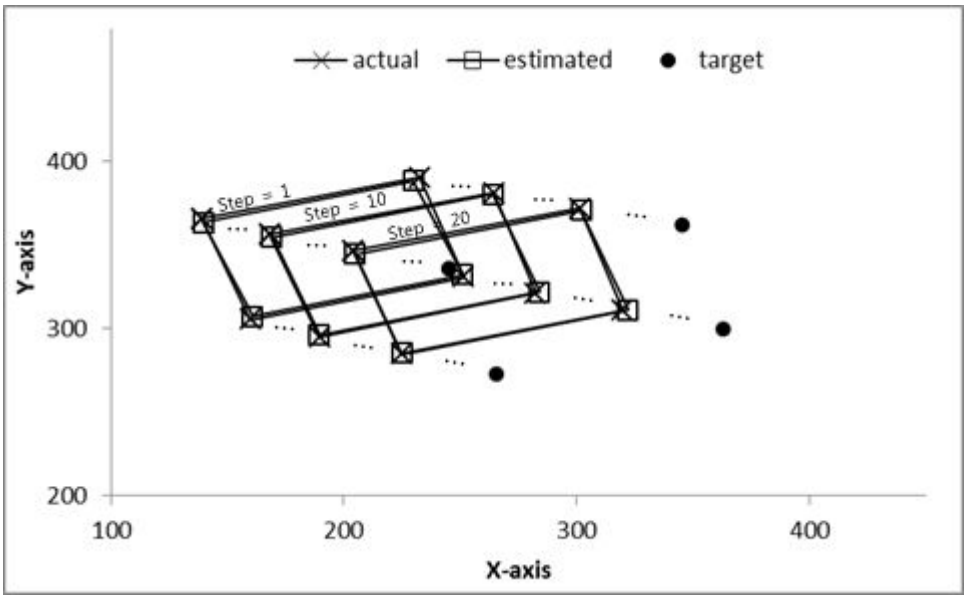
Table 6-8 The estimated six parameters using the EKF method

Step	C	Camera 1	Camera 2	Camera 3
1	C1	0.6207	0.5310	0.4586
	C2	0.1756	0.1017	0.0749
	C3	0.8584	0.8612	0.9110
	C4	1.0122	1.0263	1.0945
	C5	1167.3884	1264.5945	1509.0844
	C6	668.0372	649.3765	562.9632
⋮				
10	C1	0.6108	0.5609	0.4675
	C2	0.2241	0.1392	0.0959
	C3	0.9199	0.8630	0.9036
	C4	0.9396	1.0403	1.1042
	C5	1141.8281	1252.5626	1500.3658
	C6	509.8448	642.7029	553.5700
⋮				
20	C1	0.6136	0.5694	0.4772
	C2	0.2504	0.1512	0.1094
	C3	0.9340	0.8657	0.9071
	C4	0.9302	1.0483	1.1111
	C5	1133.2365	1255.9944	1503.6337
	C6	455.5545	637.5692	547.6237
⋮				
30	C1	0.6216	0.5792	0.4880
	C2	0.2712	0.1680	0.1235
	C3	0.9517	0.8708	0.9139
	C4	0.9153	1.0526	1.1155
	C5	1124.6804	1250.8598	1506.1143
	C6	405.4451	625.5113	538.3092

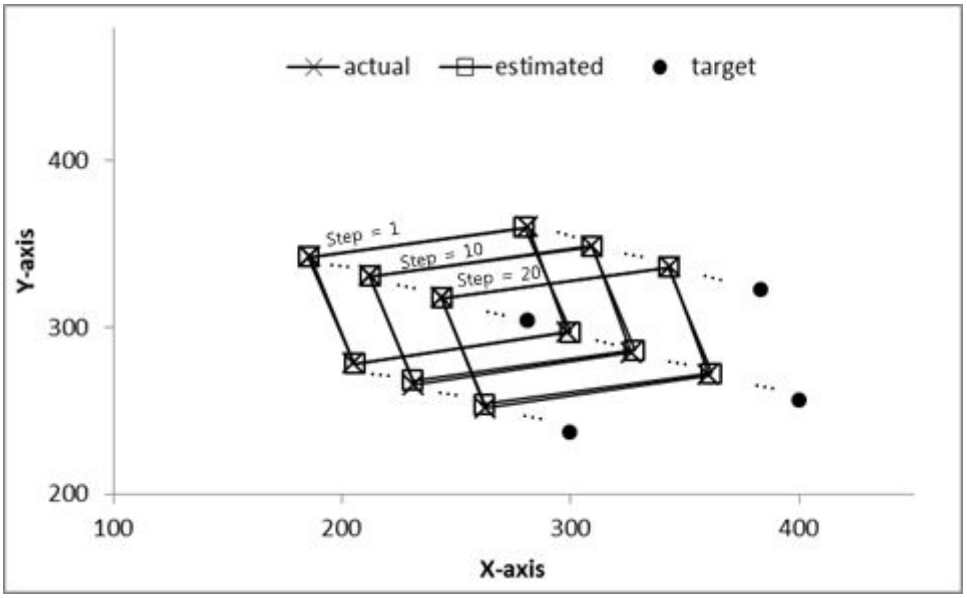
Fig. 6-8 shows the results of comparing the calculated vision system model values by applying the parameters listed in Table 6-8 to Eq. (2-20) with the obtained actual vision data by the three cameras. Fig. 6-9 shows the calculated r.m.s. error values at each point of the robot movement stage by applying Eq. (6-1). Furthermore, Table 6-9 lists that the calculated average error values in each camera by applying Eq. (6-2) correspond to 0.9313 *pixel* in camera 1, 1.1782 *pixel* in camera 2, and 1.0440 *pixel* in camera 3. This corresponds to lowest error among the three methods investigated in the study.



(a) camera 1



(b) camera 2



(c) camera 3

Fig. 6-8 Comparison of the actual vision data and estimated vision system model in EKF method

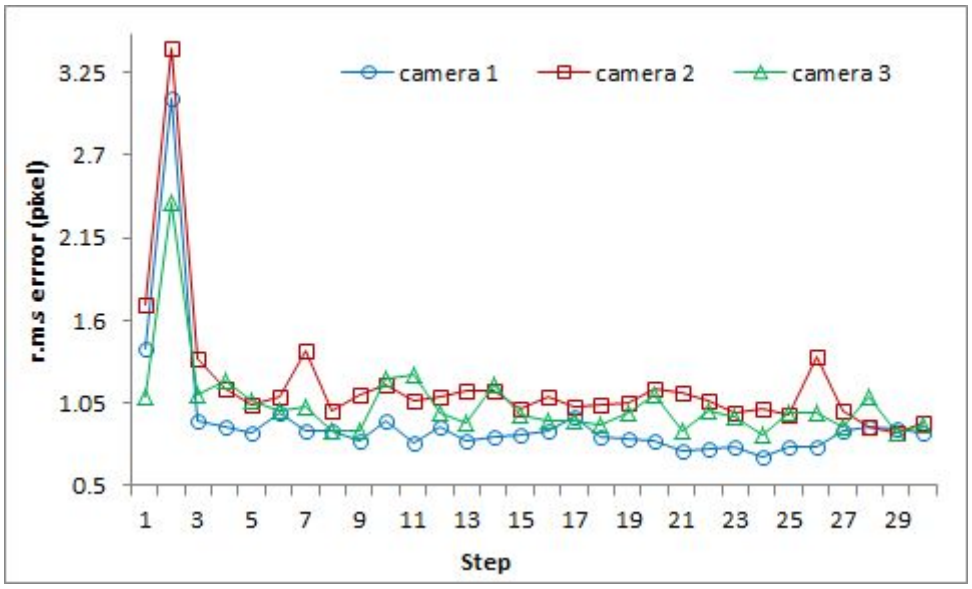


Fig. 6-9 The r.m.s. errors between actual vision data and estimated vision system model in EKF method

Table 6-9 Average errors of each camera between actual vision data and estimated vision system model in EKF method (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
Average error	0.9313	1.1782	1.0440

6.4 Comparison of vision system model's suitability

Fig. 6-10 and Table 6-10, respectively, show and list the average error of the calculated vision system model values by the three control schemes and the actual vision data by the camera, based on all vision data of the robot movement stage in each camera. The lowest error value in all the cameras is observed in the EKF method. When the N-R method without the weighting matrix and the N-R method with the weighting matrix are compared, a lower error value is obtained in the case of the N-R method without the weighting matrix.

Table 6-10 Average errors of vision system model in N-R method without weighting matrix, N-R method with weighting matrix, and EKF method using total vision data (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
N-R(no weighting)	1.8844	1.7189	1.5807
N-R(weighting)	1.9302	2.1966	2.0389
EKF	0.9313	1.1782	1.0440

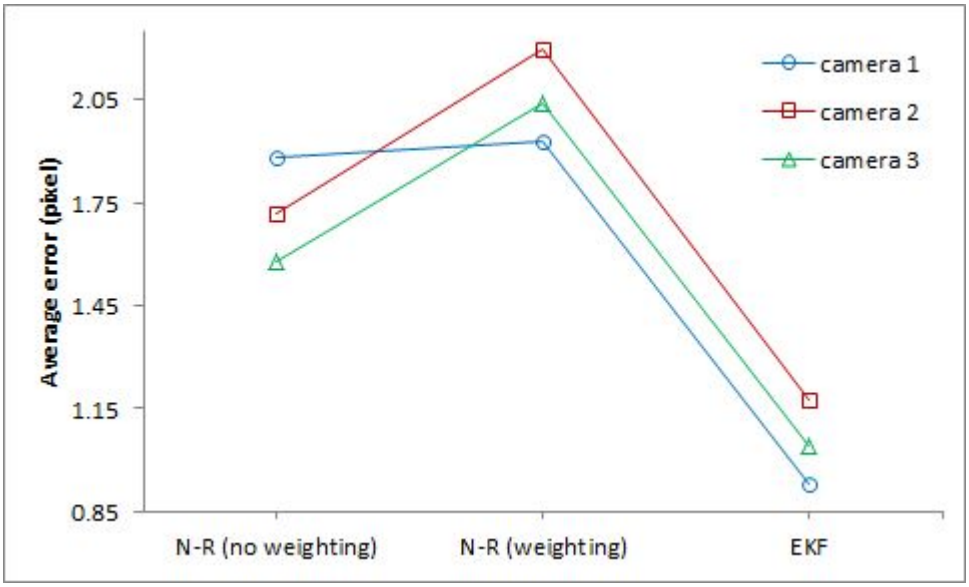


Fig. 6-10 Average errors of vision system model in N-R method without weighting matrix, N-R method with weighting matrix, and EKF method using total vision data

However, based on the recent ten data close to target, the calculated error values are shown in Table 6-11 and Fig. 6-11. Error values corresponding to 1.9646 *pixel* in camera 1, 1.7424 *pixel* in camera 2, and 1.6112 *pixel* in camera 3 are calculated when the weight matrix is not applied. In contrast, error values corresponding to 1.5492 *pixel* in camera 1, 1.5749 *pixel* in camera 2 and 1.3322 *pixel* in camera 3 are calculated when the weight matrix is applied. Thus, the results indicates that better error values are obtained when the weighting matrix is not applied. In conclusion, with respect to determining the suitability of the vision system model, the results indicates that both cases had the suitable accuracy for the fixed target estimation.

It may be noted that the error value for the EKF method is calculated to compare the error values of the N-R method without the weighting matrix and the N-R method with the weighting matrix.

Table 6-11 Average errors of vision system model in N-R method without weighting matrix, N-R method with weighting matrix, and EKF method using the recent ten data (unit: *pixel*)

	Camera 1	Camera 2	Camera 3
N-R(no weighting)	1.9646	1.7424	1.6112
N-R(weighting)	1.5492	1.5749	1.3322
EKF*	0.7929	1.0158	0.9378

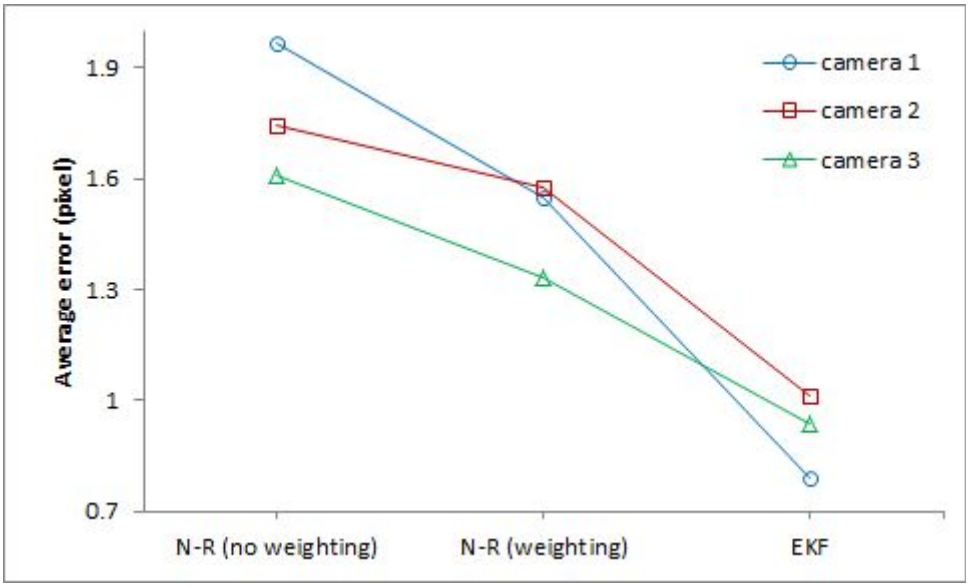


Fig. 6-11 Average errors of vision system model in N-R method without weighting matrix, N-R method with weighting matrix, and EKF method using the recent ten data

Chapter 7. Results of fixed target estimation

In this study, four methods including robot kinematic analysis, N-R method without weighting matrix, N-R method with weighting matrix, and EKF method are used to estimate the position value for the fixed target. The robot kinematic analysis calculated the robot joint angle and position value using the actual value of the target when the spatial coordinate system for the target is known. The other three methods estimate the robot joint angle and position value by only using vision data when the spatial coordinate system for the target is unknown.

The robot joint angles and position values calculated using the robot kinematic analysis are compared with the estimated joint angles and position values using the N-R method without the weighting matrix, the N-R method using the weighting matrix, and the EKF method. The error value obtained by comparing the estimated target position value with the actual target position value is defined by using r.m.s.[27] as shown in Eq. (7-1). In this equation, the estimated position values for the target are calculated by applying the joint angles calculated by the joint angle estimation model to the robot kinematics model of Eq. (2-7).

$$e_{r.m.s} = \sqrt{\frac{\sum_{j=1}^{j^*}\left\{(e_x^j)^2 + (e_y^j)^2 + (e_z^j)^2\right\}}{3 \times j^*}} \tag{7-1}$$

where, e_x^j,e_y^j and e_z^j represent an error of x value, y value, and z value for each cue (j=1 ~ 4), and j^* represents the number of cues.

7.1 Robot kinematics analysis

With repect to the rigid body placement task, the position of the target is set cue 1 to $F_x = 660\ mm$, $F_y = 60\ mm$ and $F_z = 160\ mm$, cue 2 to $F_x = 660\ mm$, $F_y = 60\ mm$ and $F_z = 130\ mm$, cue 3 to $F_x = 630\ mm$, $F_y = 96\ mm$ and $F_z = 160\ mm$, and cue 4 to $F_x = 630\ mm$, $F_y = 96\ mm$ and $F_z = 130\ mm$. The estimated results of the robot joint angles using the inverse kinematic model in section 2.2 correspond to as $\theta_1 = 5.5177^\circ, \theta_2 = 3.5822^\circ, d_3 = 124\ mm$ and $\theta_4 = -9.1011^\circ$ as listed in Table 7-1.

Table 7-1 Actual robot's joint angles to target position in robot's kinematic analysis

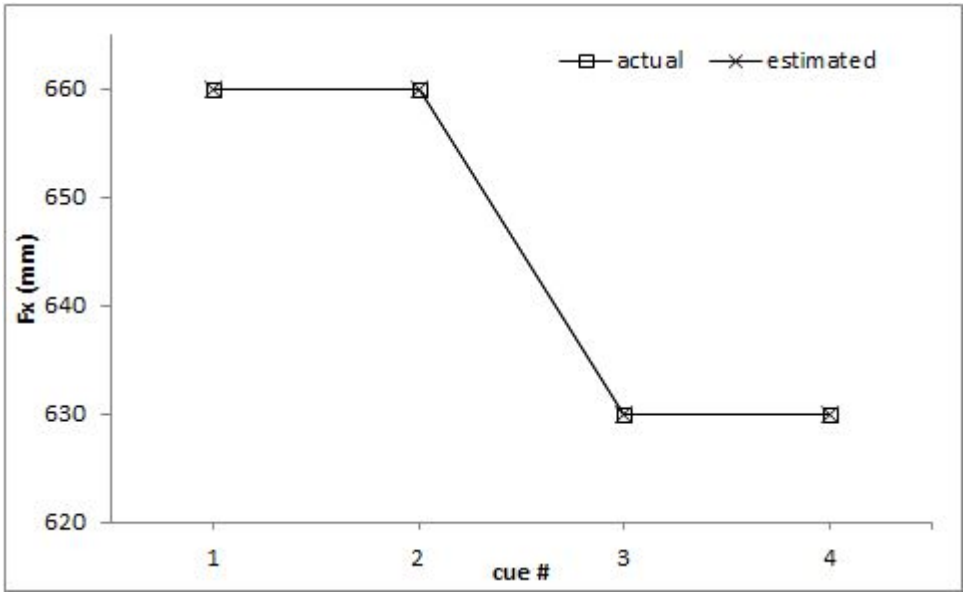
	Θ_1 (°)	Θ_2 (°)	d_3 (mm)	Θ_4 (°)
Actual	5.5177	3.5822	124	-9.1011

Table 7-2 lists the actual position values and estimated position values by applying the joint angles in Table 7-1 to Eq. (2-6). The r.m.s. error value calculated by using Eq. (7-1) corresponds to 0.0008 *mm*, which almost are similar to the actual value. This indicates that the robot kinematic model is suitable. Fig.7-1 graphically shows the difference between the actual position value and the estimated position value for F_x, F_y and F_z in each cue.

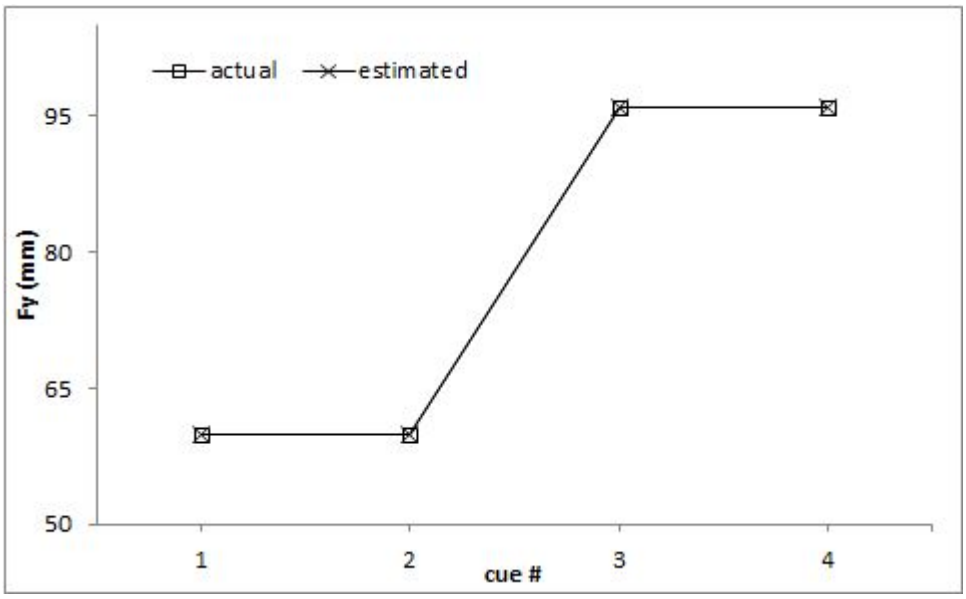
In the robot kinematic analysis, the results of the estimated robot joint angles are used as reference values. Hence, these joint angles are set as the actual joint angles to evaluate the robot joint angles of the target estimated by using the N-R method and the EKF method based on only the vision data obtained through the camera without information on the spatial position of the target.

Table 7-2 Comparison of the actual and estimated target's position values in robot's kinematic analysis

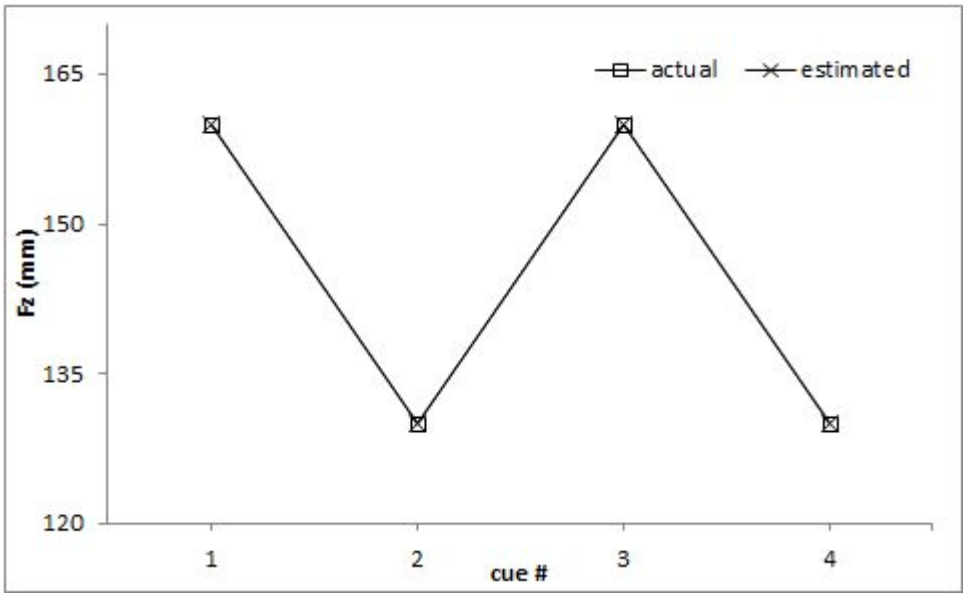
cue No.	x-y-z value	Actual	Estimated	Error (*mm*)	r.m.s. error (*mm*)
1	F_x	660	659.999	0.001	0.0008
	F_y	60	60.001	-0.001	
	F_z	160	160	0	
2	F_x	660	659.999	0.001	
	F_y	60	60.001	-0.001	
	F_z	130	130	0	
3	F_x	630	630.001	0.001	
	F_y	96	96.001	0.001	
	F_z	160	160	0	
4	F_x	630	630.001	0.001	
	F_y	96	96.001	0.001	
	F_z	130	130	0	



(a) F_x



(b) F_y



(c) F_z

Fig. 7-1 Comparison of the actual and estimated values based on the robot's kinematic analysis

7.2 Fixed target estimation using N-R method

7.2.1 Without weighting matrix

When the weighting matrix is not applied, the errors between the actual joint angles and the joint angles estimated by the robot joint angle estimation model are calculated as 0.714° in θ_1, -1.899° in θ_2, -1.139 *mm* in d_3, and -1.184° in θ_4 as shown in Table 7-3.

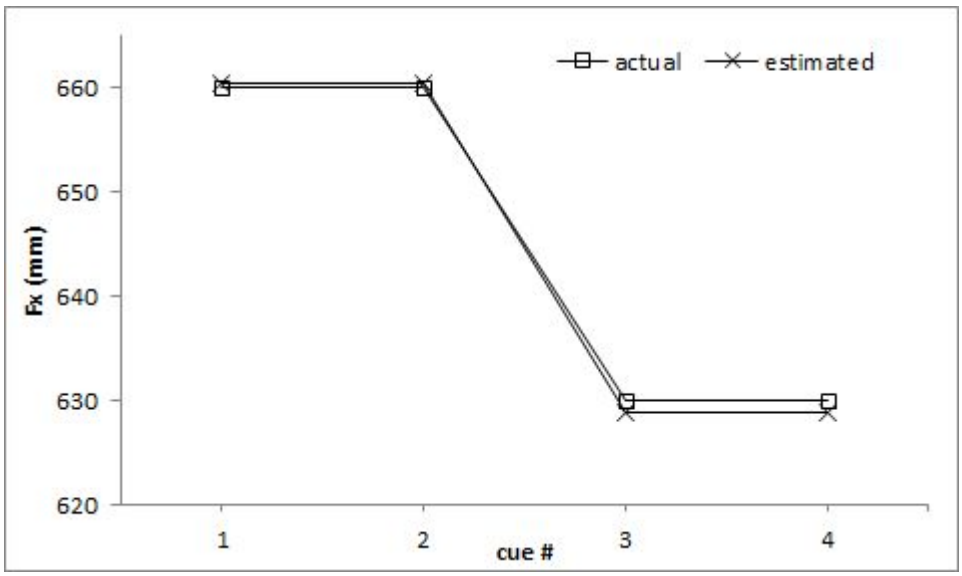
Table 7-3 Comparison of actual and estimated robot's joint angles to target in N-R method without weighting matrix

	Θ_1 (°)	Θ_2 (°)	d_3 (*mm*)	Θ_4 (°)
Actual	5.517	3.582	124	-9.101
Estimated	4.803	5.481	125.139	-7.917
error	0.714	-1.899	-1.139	-1.184

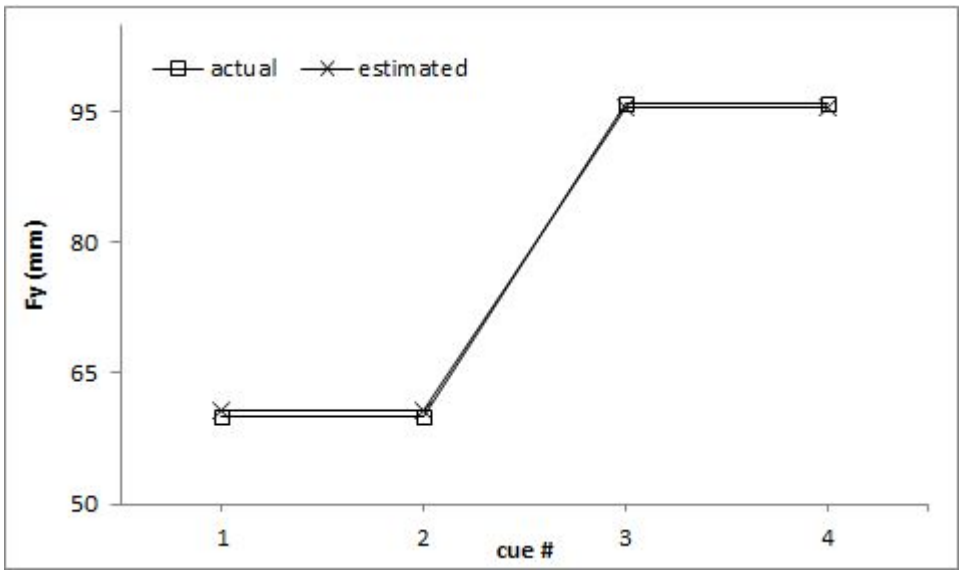
Table 7-4 shows the actual position values and the position values obtained by applying the joint angles in Table 7-3 to Eq. (2-6). The calculated r.m.s. error value by using Eq. (7-1) corresponds to 0.8995 *mm*. The processing time is measured to be less than 1 *ms*, which is measured in the program used in this study. Fig.7-1 graphically shows the difference between the actual position value and the estimated position value for F_x,F_y and F_z in each cue.

Table 7-4 Comparison of the actual and estimated target's position values in N-R method without weighting matrix

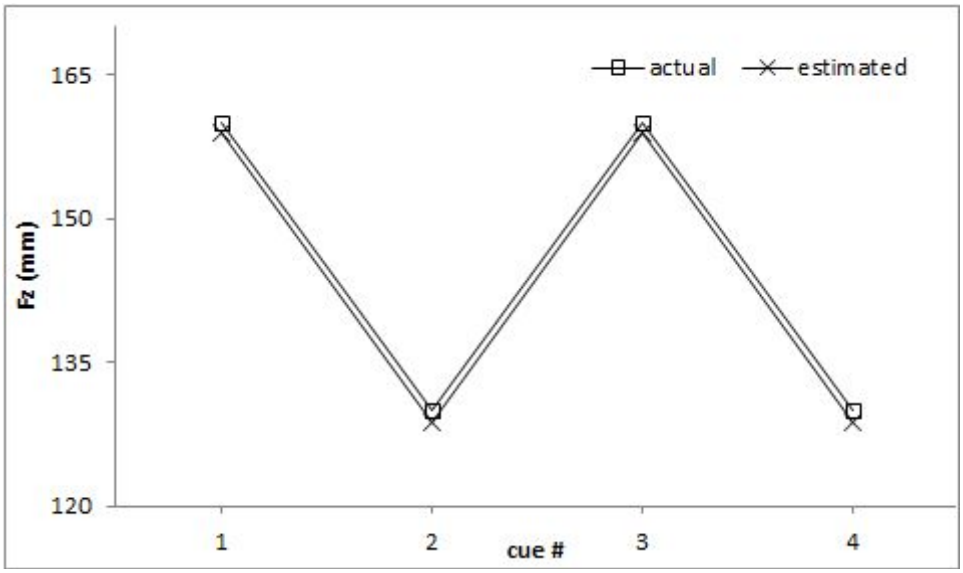
cue No.	x-y-z value	Actual	Estimated	Error (*mm*)	r.m.s. error (*mm*)	Time (*ms*)
1	F_x	660	660.31	-0.310	0.8995	*
	F_y	60	60.759	-0.759		
	F_z	160	158.861	1.138		
2	F_x	660	660.31	-0.310		
	F_y	60	60.759	-0.759		
	F_z	130	128.861	1.138		
3	F_x	630	628.848	1.152		
	F_y	96	95.489	0.511		
	F_z	160	158.861	1.138		
4	F_x	630	628.848	1.152		
	F_y	96	95.489	0.511		
	F_z	130	128.861	1.138		



(a) F_x



(b) F_y



(c) F_z

Fig. 7-2 Comparison of the actual and estimated target's position values in N-R method without weighting matrix

7.2.2 With weighting matrix

When the weighting matrix is applied, the errors between the actual joint angles and the joint angles estimated by the robot joint angle estimation model are calculated as 0.468° in θ_1, -1.172° in θ_2, -1.510 *mm* in d_3, and -0.13° in θ_4 as listed in Table 7-5. Thus, errors of the calculated joint angles are better than those of case when the weighting matrix is not applied as listed in Table 7-3.

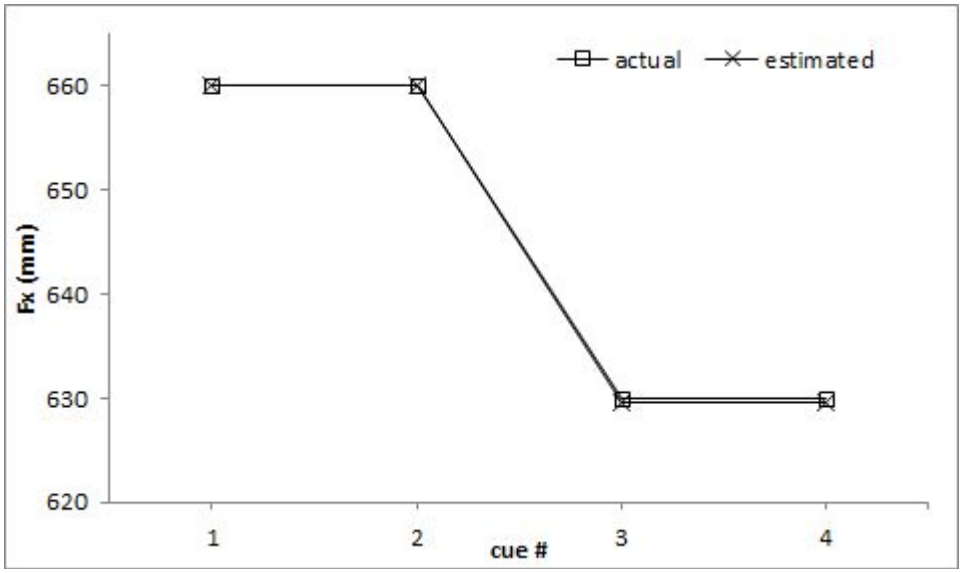
Table 7-5 Comparison of actual and estimated robot's joint angles to target in N-R method with weighting matrix

	Θ_1 (°)	Θ_2 (°)	d_3 (*mm*)	Θ_4 (°)
Actual	5.517	3.582	124	-9.101
Estimated	5.049	4.754	124.510	-8.971
error	0.468	-1.172	-0.510	-0.13

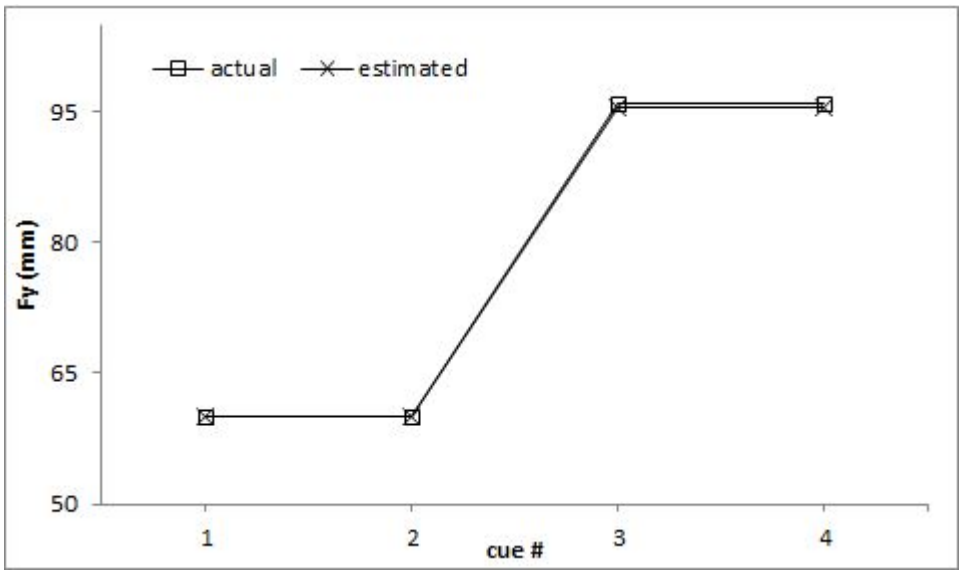
Table 7-6 lists the actual position values and the position values obtained by applying the joint angles in Table 7-5 to Eq. (2-6). The r.m.s. error value calculated by using Eq. (7-1) corresponds to 0.3965 *mm*. Thus, a better value is obtained when the weight matrix is applied. The processing time corresponds to 32 *ms*. Fig.7-3 graphically shows the difference between the actual position value and the estimated position value for F_x,F_y and F_z in each cue.

Table 7-6 Comparison of the actual and estimated target's position values in N-R method with weighting matrix

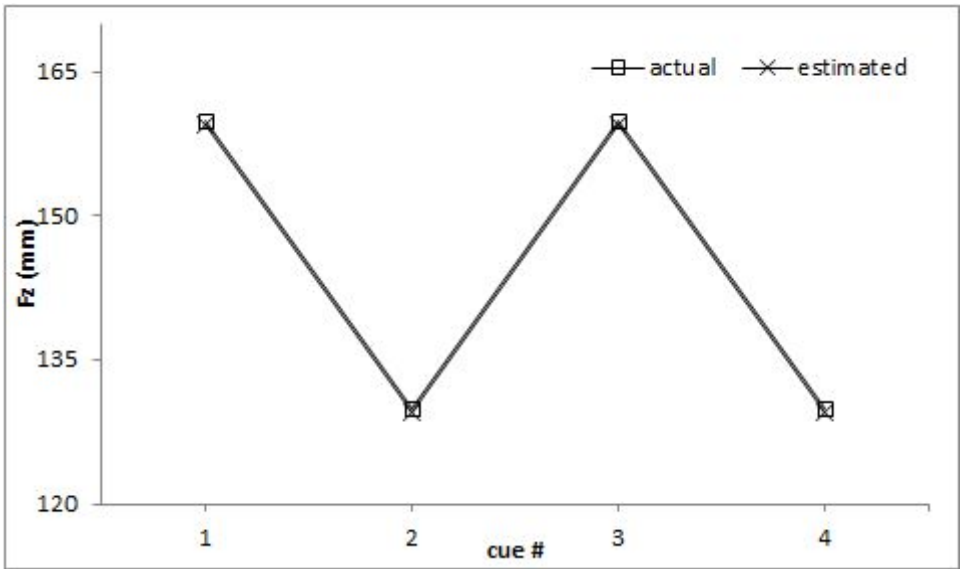
cue No.	x-y-z value	Actual	Estimated	Error (*mm*)	r.m.s. error (*mm*)	time (*ms*)
1	F_x	660	660.058	-0.058	0.3965	32
	F_y	60	59.984	0.015		
	F_z	160	159.490	0.509		
2	F_x	660	660.058	-0.058		
	F_y	60	59.984	0.015		
	F_z	130	129.490	0.509		
3	F_x	630	629.539	0.461		
	F_y	96	95.545	0.455		
	F_z	160	159.490	0.509		
4	F_x	630	629.539	0.461		
	F_y	96	95.545	0.455		
	F_z	130	129.490	0.509		



(a) F_x



(b) F_y



(c) F_z

Fig. 7-3 Comparison of the actual and estimated target's position values in N-R method with weighting matrix

7.3 Fixed target estimation using EKF method

The calculated errors between the actual joint angles and the joint angles estimated by the robot joint angle estimation model correspond to as 0.288° in θ_1, -0.716° in θ_2, -0.331 *mm* in d_3, and 0.381° in θ_4 as listed in Table 7-7. Specifically, $\theta_1 \sim d_3$ indicate a lower error values when compared with the two cases of the N-R method as listed in Table 7-3 and Table 7-5. Only the θ_4 exhibits a higher error value than the value in the N-R method with the weighting matrix. Thus, the results indicate that the EKF method is the most efficient method among the presented three control schemes.

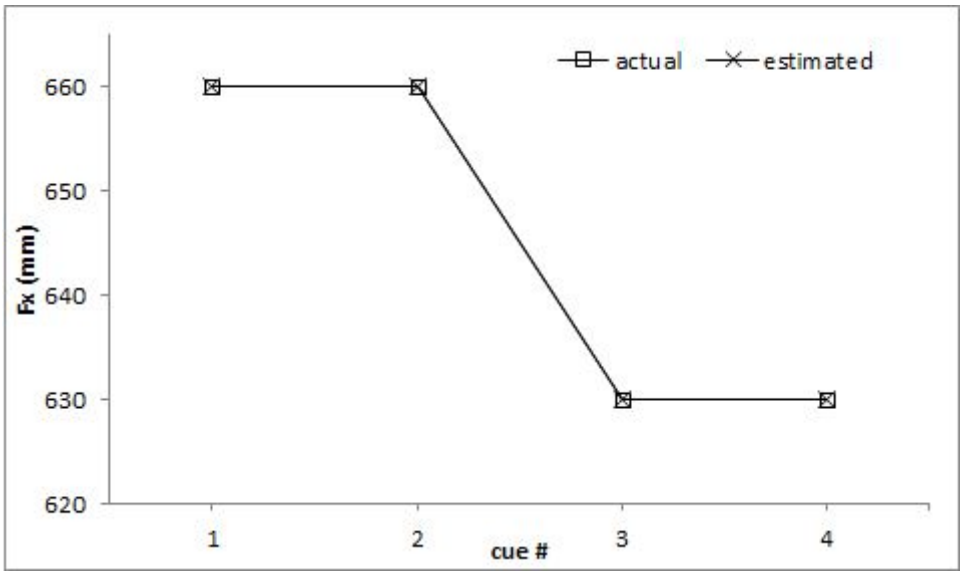
Table 7-7 Comparison of actual and estimated robot's joint angles to target in EKF method

	Θ_1 (°)	Θ_2 (°)	d_3 (*mm*)	Θ_4 (°)
Actual	5.517	3.582	124	-9.101
Estimated	5.229	4.298	124.331	-9.482
error	0.288	-0.716	-0.331	0.381

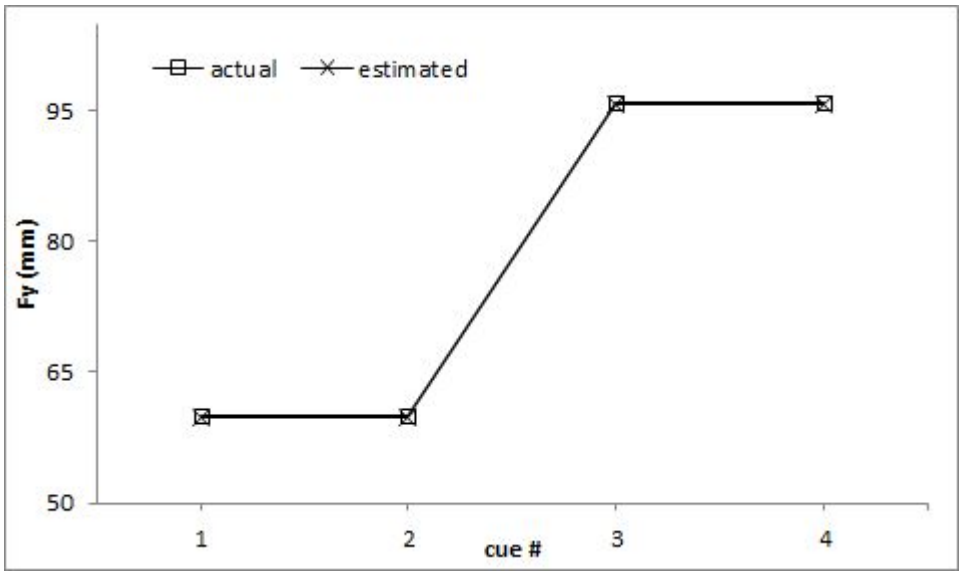
Table 7-8 lists the actual position values and the position values obtained by applying the joint angles in Table 7-7 to Eq. (2-6). The r.m.s. error value calculated by using Eq. (7-1) corresponds to 0.2241 *mm*. This corresponds to the best value among the presented three methods. The processing time is 281 *ms*. Fig. 7-4 graphically depicts the difference between the actual position value and the estimated position value for F_x,F_y and F_z in each cue.

Table 7-8 Comparison of the actual and estimated target's position values in EKF method

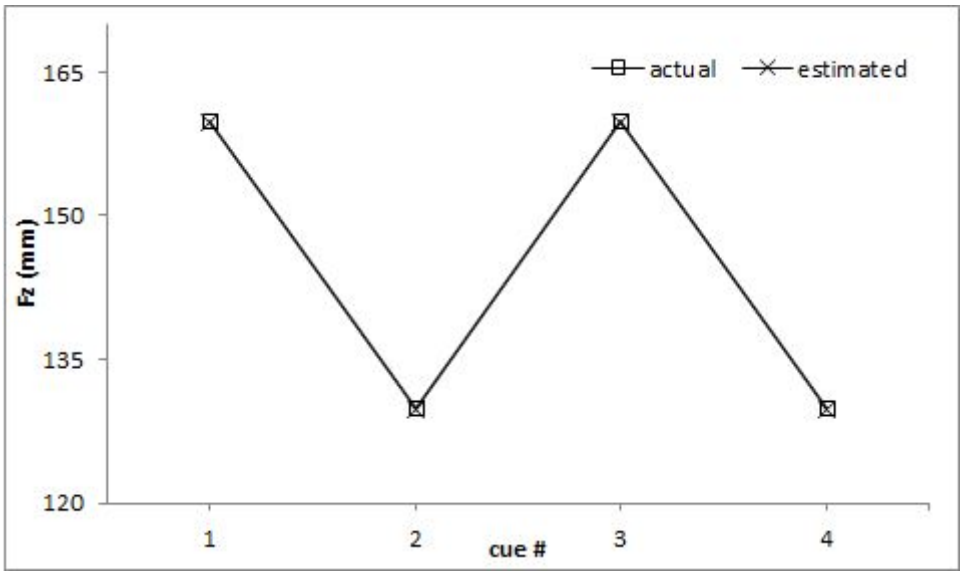
Cue No.	x-y-z value	Actual	Estimated	Error (mm)	r.m.s. error (*mm*)	time (*ms*)
1	F_x	660	659.902	0.098	0.2245	281
	F_y	60	59.844	0.156		
	F_z	160	159.669	0.331		
2	F_x	660	659.902	0.098		
	F_y	60	59.844	0.156		
	F_z	130	129.669	0.331		
3	F_x	630	629.873	0.127		
	F_y	96	95.820	0.18		
	F_z	160	159.669	0.331		
4	F_x	630	629.873	0.127		
	F_y	96	95.820	0.18		
	F_z	130	129.669	0.331		



(a) F_x



(b) F_y



(c) F_z

Fig. 7-4 Comparison of the actual and estimated target's position values in EKF method

7.4 Comparison of fixed target estimation

Table 7-9 shows the results of the N-R method without the weighting matrix, N-R method with the weighting matrix, and EKF method in order to compare with the robot joint angle for the target calculated by the robot kinematic analysis in chapter 2.

In almost all cases, the values calculated by the EKF method are closest to the actual values. Especially, in N-R method, the values calculated with the weighting matrix are closer to the actual values than values calculated without the weighting matrix.

Table 7-9 The estimated robot's joint angle in kinematic analysis and three vision control scheme

Method	$\theta_1(^\circ)$	$\theta_2(^\circ)$	$d_3(mm)$	$\theta_4(^\circ)$
Kinematics analysis	5.517	3.582	124	-9.101
N-R (no weighting)	4.803	5.481	125.139	-7.917
N-R (weighting)	5.049	4.754	124.510	-8.971
EKF	5.229	4.298	124.331	-9.482

The calculated joint angles by each control scheme are applied to the kinematic model, and the calculated position values of the fixed target are compared with the actual target position as shown in Table 7-10. In a manner similar to the results obtained in the cases of the joint angle, the position values of the fixed target calculated by the EKF method are closest to the actual values in almost all cases. Furthermore, the values calculated by the N-R method with the weighting matrix are closer to the actual values than those of the N-R method without the weighting matrix.

Table 7-10 Comparison of target's position value in kinematic analysis and three vision control scheme (*mm*)

Cue	x-y-z value	Actual	Kinematics analysis	N-R (no weighting)	N-R (weighting)	EKF
1	F_x	660	659.999	660.31	660.058	659.902
	F_y	60	60.001	60.759	59.984	59.844
	F_z	160	160	158.861	159.490	159.669
2	F_x	660	659.999	660.31	660.058	659.902
	F_y	60	60.001	60.759	59.984	59.844
	F_z	130	130	128.861	129.490	129.669
3	F_x	630	630.001	628.848	629.539	629.873
	F_y	96	96.001	95.489	95.545	95.820
	F_z	160	160	158.861	159.490	159.669
4	F_x	630	630.001	628.848	629.539	629.873
	F_y	96	96.001	95.489	95.545	95.820
	F_z	130	130	128.861	129.490	129.669

Fig.7-5 and Table 7-11, respectively, show and list the r.m.s. error value calculated by applying the estimated position value of the fixed target in Table 7-10 to Eq. (7-1). Kinematic analysis method with an r.m.s. error of 0.0008 *mm* is used as reference method to evaluate the effectiveness of three proposed control schemes. This indicates that among three presented control schemes, the best results are obtained in the case of the EKF method and corresponds to 0.2241 *mm*. Additionally, the r.m.s. error of 0.3695 *mm* obtained in the case of the N-R method with the weighting matrix, exceeds 0.8995 *mm* in the case of the N-R method without the weighting matrix. Finally, the results indicate that among three proposed control schemes, the EKF method exhibits the best accuracy with respect to fixed target estimation for rigid body placement.

Table 7-11 The r.m.s. errors in kinematic analysis and three vision control scheme (*mm*)

	Kinematics analysis	N-R (no weighting)	N-R (weighting)	EKF
r.m.s. error	0.0008	0.8995	0.3965	0.2245

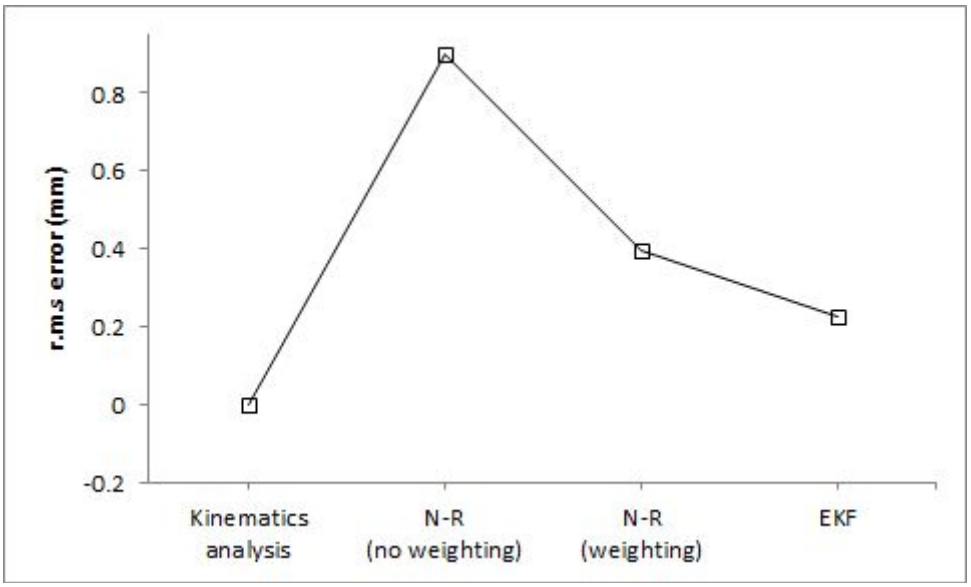


Fig. 7-5 Comparison of the r.m.s. errors in kinematic analysis and three vision control scheme

Chapter 8. Summary and conclusions

This study is concerned with estimating a fixed target position in cases where the spatial position is unknown for a rigid body placement task, while moving a robot towards the target. Hence, three real-time robot vision control schemes are proposed to efficiently process the obtained vision data, and they include the N-R method without the weighting matrix and the N-R method with the weighting matrix, and the EKF method. Additionally, the calculated result from the robot kinematics analysis is set as a standard in the cases, when the spatial position of the target is known. The three proposed real time control schemes are applied to a fixed target estimation experiments in which the position value of the spatial target for the rigid body placement task is unknown. The results of robot kinematic analysis are used to compare the estimated values of the fixed target position as the reference values, and the effectiveness of the three proposed control schemes are compared. The results are summarized as follows.

8.1 Summary of results

(1) Development of the weighting matrix model

① In the N-R method, a weighted matrix model is proposed to place weights on the most recent acquired data near the target while the robot moved towards the target.

② The proposed weighting matrix model is used to estimate camera parameters with improved accuracy.

③ Based on the proposed weighting matrix model, 0.17 is calculated as the optimal weighting factor to create a new weighting matrix ,W, and used to estimate the target value in the robot movement stage.

(2) Suitability of the vision system model

The results using three vision control schemes are as follows.

- In the N-R method without the weighting matrix, the average error value for each camera is calculated as 1.8844 *pixel* in camera 1, 1.7189 *pixel* in camera 2, and 1.5807 *pixel* in camera 3. This corresponded to an average value of 1.728 *pixel*.
- In the N-R method with the weighting matrix for all data, the average error value for each camera is calculated as 1.9302 *pixel* in camera 1, 2.1966 *pixel* in camera 2, and 2.0389 *pixel* in camera 3. This corresponded to an average value of 2.0552 *pixel*. In contrast, in the N-R method with the weighting matrix for recent ten data points, the average error value for each camera is calculated as 1.5492 *pixel* in camera 1, 1.5749 *pixel* in camera 2, and 1.3322 *pixel* in camera 3. This corresponded to an average value of 1.4854 *pixel*.
- In the EKF method, the average error value for each camera is calculated as 0.9313 *pixel* in camera 1, 1.1782 *pixel* in camera 2, and 1.0440 *pixel* in camera 3. This corresponded to an average value of 1.0512 *pixel*.

The results of the suitability of the fore-mentioned vision system model are summarized as follows.

① The suitability of the vision system model in the three control schemes is compared. The results indicated that the lowest error value is calculated in the EKF method.

② The suitability of the vision system model in the cases of the N-R method is compared, and the error value is calculated when the acquired all data used. The results indicate that a lower error value is calculated when the N-R method without weighting matrix is used. In contrast, when the recent ten data near the target is used, a lower error value is calculated using the N-R method with the weighting matrix than that in the case of the N-R method without the weighting matrix.

(3) Fixed target estimation

Comparing using the actual position value with the estimated position value, the r.m.s. error values are calculated as follows.

- In the robot kinematic analysis, the r.m.s. error value is calculated as 0.0008 *mm*.
- In the N-R method without the weighting matrix, the r.m.s. error value is calculated as 0.8995 *mm*. Additionally, the processing time is measured as less than 1 *ms*, which could be measured through the program used in this study.
- In the N-R method with the weighting matrix, the r.m.s. error value is calculated as 0.3965 *mm*. The processing time is measured as 32 *ms*.
- In the EKF method, the r.m.s. error value is calculated as 0.2245 *mm*. The processing time is measured as 281 *ms*.

The fixed target estimation results shown above are summarized as follows.

① The calculated error values from the robot kinematics analysis are almost equal to the actual values, and thus, indicate the suitability of the robot kinematic model. Hence, these are set as the reference values in order to evaluate the three proposed vision control schemes.

② The estimated robot joint angles for the target through robot kinematics analysis is acquired by using a known spatial position of the target. Thus, the calculated joint angles through the robot kinematics analysis are considered as the actual joint angles. These results are used to compare the estimated robot joint angles to the target through the three proposed control schemes with the vision data obtained only through the camera without the information on the spatial position of the target.

③ The r.m.s. error between the actual position and the estimated position value calculated in the EKF method exhibit the best accuracy among the three proposed control schemes. Furthermore, the comparison of the two cases of the N-R method indicates that the N-R method with the weighting matrix had a better accuracy than the N-R method without the weighting matrix

8.2 Conclusions

The conclusions regarding the suitability of the robot vision system model and fixed target estimation using the three proposed real time robot vision control schemes for fixed target estimation for rigid body placement task include the followings.

(1) The results with respect to the suitability of the vision system model indicate that the model exhibited good accuracy for fixed target estimation in the rigid body placement task in the three proposed robot control schemes.

(2) The estimated target position values by using the robot kinematic analysis, which is used as the reference values to evaluate the effectiveness of the three proposed control schemes, are almost equal to the actual target values. Thus, this shows the validity of the robot kinematic model.

(3) With respect to only accuracy, the results suggest that the EKF method is the most effective among the three proposed robot vision control schemes. Additionally, the N-R method with the weighting matrix involving the optimal weighting factor exhibits better accuracy than the N-R method without the weighting matrix. This indicates the validity of the weighting matrix model.

(4) With respect to both precision and the processing times, the N-R method with the weighting matrix is more efficient than the EKF method.

This study presents the robot vision control schemes to estimate a fixed target for rigid body placement task. Future studies will investigate a robot vision control scheme to estimate a moving target.

REFERENCES

(1) Feddema, J. T., Lee, C. S. G., "Adaptive Image Feature Prediction and Control for Visual Tracking with a Hand-Eye Coordinated Camera", IEEE Transactions on system, man and cybernetics, Vol. 20, No. 5, pp.1172~1183, 1990.

(2) Assa, A., Janabi-Sharifi, F., "A Robust Vision-Based Sensor Fusion Approach for Real-Time Pose Estimation", IEEE transactions on cybernetics, Vol. 44, No. 2, pp.217~227, 2014.

(3) Lowe, D. G., "Fitting Parameterized Three-Dimensional Models to Images", IEEE transaction on pattern analysis and machine., Vol. 13, No. 5, pp.441~450, 1991.

(4) Jahari, M., Yamamoto, K., Miyamoto, M., Kondo, N., Ogawa, Y., Suzuki, T., Habaragamuwa, H., and Ahmad. U., "Double Lighting Machine Vision System to Monitor Harvested Paddy Grain Quality during Head-Feeding Combine Harvester Operation", journal of ISSN machines, Vol. 3, pp.352~363, 2015.

(5) Huang, K. Y., Ye, Y. T., "A Novel Machine Vision System for the Inspection of Micro-Spray Nozzle", journal of ISSN sensors, Vol. 15, pp.15326~15338, 2015.

(6) Hosoda, K., Kamado, M., and Asada, M., "Vision Based Servoing Control for Legged Robots", Proceedings of the 1997 IEEE., pp.3154~3159, 1997.

(7) Choe, Y. E., Lee, H. C., Kim, Y. J., Hong, D. H., Park, S. S., and Lim, M. T., "Vision Based Estimation of Bolt-Hole Location using Circular Hough Transform", ICROC-SICE International conference August pp.4821~4826, 2009.

(8) Asl, H. J., Bolandi, H., "Robust Vision based Control of an Underactuated Flying Robot Tracking a Moving Target", transaction of the Institute of Measurement Control, pp.1~14, 2013.

(9) Oh, H. C., Kang, T. K., Bea, D. S., Yu, S. Y., Joen, T. J., Koehler, C. G.

S., Park, S. K., and Lim, M. T., "Object Recognition and a Robot Arm Control using a 2D Camera", proceeding of ICROS, pp.107~108, 2014.

(10) Kim, K. T., Seo, K. S., "Evolutionary Generation based Color Detection Technique for Object Identification in Degraded Robot Vision" Transactions of KIEE Vol. 64, No. 7, pp.1040~1046, 2015.

(11) Choi, C. S., Lee, J. Y., Kim, Y. L., "Initial Point Optimization for Square Root Approximation based on Newton-Raphson Method" journal of IEEK Vol. 43-SD, No. 3, pp.15~20, 2006.

(12) Bae, M. J., "Extraction of Slot Parameters for Waveguide Antennas Using the Newton Raphson Method" journal of KIIT Vol. 12, No. 5, pp.33~41, 2014.

(13) Durmus, B., Temurtas, H., Yumusak, N., Temurtas, T., Kazan, R., "The Cost Function Minimization for Predictive Control by Newton-Raphson Method", proceeding of the international multiconference of engineers and computer scientists, Vol. 2, pp. 19~21, 2008.

(14) Yang, C., Huang, Q., Ogbobe, P. O., Han, J., "Forward Kinematics Analysis of Parallel Robust using Global Newton-Raphson Method", proceeding of international conference on intelligent computation technology and automation, pp.407-410, 2009.

(15) Jung, Y. J., Kim, G. W., "Nonlinear Least Squares Method and The Normalized Weighting Matrix Navigation Algorithm for Mobile Robots Using", proceeding of conference on CICS, pp. 70~71, 2013.

(16) Kim, D. H., Lee, D. H., Myung, H., Choi, H. T., "Vision based Localization for AUVs using Weighted Template Matching in a Structured Environment", journal of institute of control robotics and systems, Vol. 19, No. 8, pp.667~675, 2013.

(17) Rigatos, G. G., Siano, P., "Sensorless Control of Electric Motors with Kalman Filters: Applications to Robotic and Industrial Systems", international journal of advanced robotic system, Vol. 8, No. 6, pp.62~80, 2011.

(18) Karasalo, M., Hu, X., "An Optimization Approach to Adaptive Kalman

Filtering", journal of automat, Vol. 47, No. 8, pp.1785~1793, 2011.

(19) Chen, G., Xia, Z., Ming, X., Lining, S., Ji, J., Du, Z., "Camera Calibration based on Extended Kalman Filter using Robot's Arm Motion", proceeding of IEEE/ASME international conference on advanced intelligent mechatronics , pp.1839~1844, 2009.

(20) Hwang, W. S., Park, J. H., Kwon, H. I., Anjum, M. L., Kim, J. H., Lee, C. H., Kim, K. S., and Cho, D. I., "Vision Tracking System for Mobile Robots Using Two Kalman Filters and a Slip Detector", international conference on control automation and systems, pp.2041~2046, 2010.

(21) Darabi, S., Shahri, M., "Iteration Effect on Vision Based Simultaneous Localization and Mapping using Kalman Filters Family", proceeding of the IEEE international conference on robotics and biomimetics, pp.1084~1089, 2011.

(22) Jeon, Y. P., "Research for Extended Kalman Filter Localization using Vision System and Improving Navigation of Indoor Mobile Robot using Vector Field Histogram Method", Thesis of Master, school of electronic engineering, chonbuk university., 2013.

(23) R. Kelly, R. Carelli, O. Nasisis, B. Kuchen, and F.Reyes, "Stable Visual servoing of camera-inhand robotics systems", IEEE/ASME Trsns. on Mechatronics, Vol. 5, no. 1, pp.39-48, Mar. 2000

(24) Yoshihiro, TODA., Yasuo, KONISHI., Hiroyuki, ISHIGAKI. "Positioning-Control of Robot Manipulator Using Visual Sensor," Int. Conference on Control, Automation, Robotics and Vision, pp.894~898, December. 1996.

(25) Berthold, K. P. H., "Robot vision", Cambridge, Massachusetts, The MIT Press., pp.46~48, 1986.

(26) Peter, A. S., "Control of eye and arm movements using active, attentional vision", Applications of AI, machine vision and robotics., pp.1471~1491, 1993.

(27) Freedman, D., Pisani, R., Purves, R., "Statistic Fourth edition", Inc W.W.Norton & Company, pp.66~67, 2007

APPENDIX

A. Monte-Carlo method

The detailed explanation for obtaining the initial values of the EKF control scheme using the Monte-Carlo method shown in Fig. A-1 is as follows.

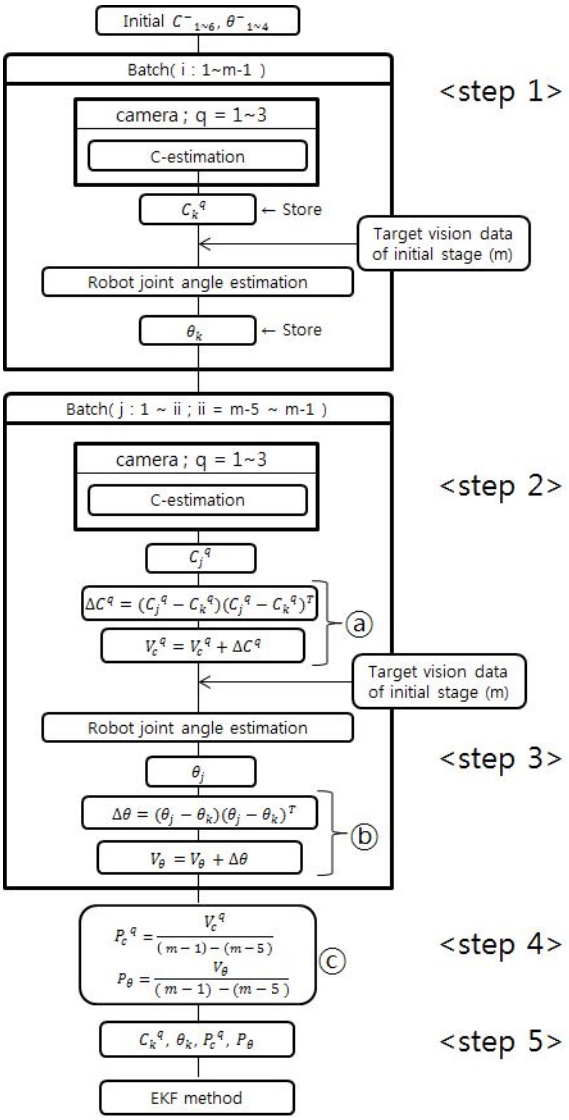


Fig. A-1 Procedures of Monte-Carlo method

① step 1

In the initial stage with m(=10) vision data, the parameters C_k^q $(k=1\sim6)$ for each camera are estimated using $(m-1)$ vision data through a batch scheme. And, the robot joint angles θ_k $(k=1\sim4)$ are estimated using C_k^q and the mth target vision data. The estimated C_k^q and θ_k are stored, and used to calculate the error covariance of the parameters for each camera and the robot joint angle.

② step 2

The parameters C_j^q $(j=1\sim6)$ for each camera are estimated using $(m-1)$ vision data. Then, calculate V_c^q by applying the stored C_j^q and C_k^q in step 1 to ⓐ of Fig. A-1.

③ step 3

For $(m-5)$ vision data, the robot joint angles θ_j $(j=1\sim4)$ are estimated using estimated parameters C_j^q for each camera in step 2 and the mth target vision data. Calculate V_θ by applying the stored θ_j and θ_k in step 1 to ⓐ of Fig. A-1. Then, step 2 and step 3 are repeated while increasing the number of vision data to $(m-1)$.

④ step 4

Calculate the error covariance of the parameter for each camera P_c^q and the error covariance of the robot joint angle P_θ by applying the estimated V_c^q and V_θ in step 2 and step 3 to ⓒ of Fig. A-1.

⑤ step 5

The calculated C_k^q for each camera parameter and θ_k for robot joint angle in step 1 as well as the calculated error covariance P_c^q for each camera parameter and the error covariance P_θ for robot joint angle in step 4 are used as the initial values of the EKF control scheme.

ACKNOWLEDGEMENTS

First of all, I would like to express the deepest appreciation to my supervisor, Professor Jang Wan-Shik, for making me do this thesis and for the knowledge that I have gained from him.

I would like the give my special thanks to Professor Jeong Sang-Wha and Professor Sung Yoon-Gyung for serving as my thesis committee members.

And, I would like to express my gratitude to Mr. Kim Jae-Myung, Mr. Hong Sung-Moon, Mr. Nam Min-Guk, and Mr. No Tae-Wan of Mechanical Control Engineering Laboratory. They are nice members and always be there to support and encourage me when I need.

Also, I wish to thank Hanguk Gym's master, Mr. Jang Chun-Suk and Hanguk Gym's friends. They are good friends and always stayed next to me when I tired.

Finally, I must express my best gratitude to my family, father, mother and sister, who have always supported me through all situations. They always provide me with unfailing supports and continuous encouragements during these years. This accomplishment would not have been possible without them. Thank you

December, 2016

yours very truly, Jang Min-Woo