'93 한국자동제어학술회의논문집 (1993.10.20~22)

시각정보에 의한 로보트 매니퓰레이터의 위치·자세 제어 (신경회로망의 이용)

심 귀 보
중앙대학교 공과대학 제어계측공학과

Visual Control of Robotic Manipulators Based on Neural Network

Kwee-Bo Sim
Dept. of Control and Instrumentation Engineering, Chung-Ang University

ABSTRACT

This paper describes a control scheme for a robot manipulator system which uses visual information to position and orientate the end-effector. In this scheme, the position and orientation of the target workpiece with respect to the base frame of the robot are assumed to be unknown, but the desired relative position and orientation of the end-effector to the target workpiece are given in advance. The control scheme directly integrates visual data into the servoing process without subdividing the process into determination of the position and orientation of the workpiece and inverse kinematics calculation. A neural network system is used for determining the change in joint angles required in order to achieve the desired position and orientation. The proposed system can be control the robot so that it approach the desired position and orientation from arbitrary initial ones. Simulation for the robot manipulator with six degrees of freedom will be done. The validity and the effectiveness of the proposed control scheme will be verified by computer simulations.

1. Introduction

In recent years, systems that integrate both visual sensors and robot together have received a lot of attention, especially in the field of intelligent robot[1][2]. Such systems can solve many problems which limit applications of current robots. Recently, many researchers have discussed possibilities for the application of neural networks in robot control. The basic theme of such discussion is that of using the network to learn the characteristics of the robot system[3].

In this paper a control scheme for a robotic manipulator with visual sensors is presented, which makes use of visual information to position and orientate the end-effector. In this scheme, the position and orientation of the target workpiece with respect to the base frame of the robot are assumed to be unknown, but the desired relative position and orientation of the end-effector to the target workpiece are given in advence. The proposed system controls the robot so that it approach the desired position and orientation from arbitary initial ones.

The control system integtates visual data into the servoing process directly without subdividing the process into determination of the position and orientation of the workpiece and inverse kinematic calculation. A neural network system is used for determining the change in joint angles required in other to achieve the desired position and orientation.

This paper is structured as follows. The general concept of the system is given and explained in Section 2. The structure, the principle, and the learning process of the neural network are explained in Section 3. The control strategy is discussed in Section 4, and the result of computer simulations are shown in Section 5.

2. General Concept of the System

2-1. System Configulation

In robotic applications, the task of pick and place is the most fundamental one for robotic manipulators. Usually this task consists of the following subtasks; guiding a robotic manipulator to a workpiece and letting it pick the workpiece and moving it to another place. In this study, the first subtask, guiding a robotic manipulator to a workpiece is discussed. To pick a workpiece, the position and orientation of the manipulator play importment roles. The manipulator must keep the desired relative position and orientation to the workpiece placed in arbitray position. This constrained motion of the manipulator removes most of the difficulties included in other subtasks such as picking a workpiece.

Figure 1 shows a visual servo control system discussed in this paper. The objective is to move the end-effector of a robotic manipulator towards a target point. The movement of the manipulator is achived by generating the control signals through the use of visual information and the neural network. A camera is mounted on the end-effector of the manipulator so that the optical axis is identical to the z-axis of the end-effector frame with some offset. If the model of the manipulator, such as the configuration and the links, is assumed to be known and the current values of joint angles can be obtained from shaft encoders, then the position and orientation of the manipulator with respect to the workpiece (or conversely, the position and orientation of the workpiece with respect to the base of the manipulator) can be calculated simply by coordinate transformations.

2-2. Control Principle

In general, the coordinate transformation of the four steps is needed in order to determine the control inputs to the joint angles from visual data. The procedure of coordinate transformation is shown in Figure 2. However, it requires for large amount of nonlinear calculation to solve step (3) and (4). Most of the resent researches pay attention to reconstructing a three dimensional shape from a two dimensional image data or to determining the relative position and orientation between the camera coordinate system and the workpiece[4]. On the other hand, there are few researchers that directly determine joint angles of manipulator from image data[5]. In this paper, a control strategy is proposed to control the manipulators so that it can approach the desired position and orientation without any coordinate transformation, which needs a lot of calculation.

In the camera picture, there are four visual cues on the target workpiece and the relation of these four cues is known in advance. With there data and the x-y coordinate values of these four points in the camera picture, the current position and orientation of the end-effector with respect to the workpiece can be uniquely determined by geometric calculation[6]. On the basis of the above theory, the following visual control method is introduced. Figure 3 shows the image received by the camera. Figure 3(a) is the current camera image and 3(b) is the final camera image when the end-effector reaches the desired relative position and orientation to the workpiece. The visual servo control system computers the servo error signals in terms of the positional coordinates derived from the visual image.

The proposed system generates the change in joint angles so that the coordinates of the four visual cues on the current image coincide with their desired one (A-A*, B-B*, C-C*, D-D*). The relation between the changes $\Delta\theta_i$ in joint angles and the changes (Δx_i, Δy_i) on the coordinateds of the four visual cues on the camera image is a highly nonlinear function. Here an artificial neural network is utilized to learn this non-linear function. The detail of the learning and structure of the neural network will be presented in the next section.

The system is designed to operate under the following conditions. There are many identical workpieces which come into the working area of the robot. A workpiece comes into a visual range of the camera, where its orientation and even position relative to the manipulator is arbitrary. Then the manipulator is required to move its end-effector to the pre-designed position and orientation relative to that workpiece. Here it is assumed that there are four visual cues on that workpiece and that the camera system is able to distinguish there cues from one and other.

3. Neural Network System

A PDP network is used as a neural network because it has a learning capability to map a set of input peterns to a set of output patterns[7]. The learning algorithm used in this network is Back Propagation.

The input of the neural network are the x-y coordinate values of the four relative positions between the current cues and the desired ones. The output of the network are the required changes in the joint angles for the manipulator. The desired x-y coordinate values of thr four visual cues in the camera picture are obtained in advance by moving the end-effector to the desired point. After the neural network learns the relation between input patterns and output patterns sufficiently, it shows a model of relation between the position and orientation of the end-effector and those of the workpiece. A good model is obtained without any knowledge about the manipulator by learning.

The structure of neural network is essentially suited for parallel processing. The execute time for information processing using the neural network is very fast compared with ones using serial processing.

4. Control Strategy

The block diagrams of the learning process is shown in figure 4. The network learn the relationship between the change inthe joint angles and the changes in the coordinate values of the four visual cues in the camera picture by asking the manipulator to the execute some random displacements from the desired position and orientation. The manipulator is set up in the desired position and orientation to the workpiece at first. In this stage, Δx_i, Δy_i, $\Delta\theta_i$ are zero. Then random joint angle inputs $\Delta\theta_i$ are provided for the manipulator and the visual cues move to different positions in the camera picture. The random joint angle inputs $\Delta\theta_i$ are teaching signals and Δx_i, Δy_i are input signals to the neural network. During learning process, weights of connections of neural are changed so that errors (differences between teaching signals $\Delta\theta_i$ and outputs of the network) decrease to zero.

In the execution process as shown in Figure 5, the network calculates the required changes in the joint angles in order to set the four cues into the desired positions in the camera picture. After these changes are executed by the manipulator, the camera picture will be checked again and above process will be repeated if necessary.

5. Simulation Study

Simulation result for the robotic manipulator with six degrees of freedom are shown in this section. The PDP network used in the simulation is a three layered network with 8 inputs, 6 outputs and 48 hidden units as shown in Figure 6. This net maps the eigth inputs characterizung the four pairs of cues onto the six outputs which arethe six control signals for change in the angular positions.

5-1. Learning Process

Desired positions of visual cues are given as

$$A^* = (192, 207) \qquad B^* = (91, 120)$$
$$C^* = (293, 120) \qquad D^* = (192, 274)$$

in the camera frame.

Random joint angles are produced under restrictions of Table 1. 60 patterns (each patterns consists of $\Delta\theta_i$ (i=1, $\cdots$, 6) are used as input patterns to the manipulator. Then, the differences (Δx_i, Δy_i) (i=1, $\cdots$, 6) between the positions of visual cues in the present camera picture and those in the previous picture (desired picture) are obtainrd by moving the manipulator. Table 2 shows some of these training patterns for learning process. These difference arefed to the inputs of the neural network. The errors $\Delta\theta_{ie}$ between the teaching signals $\Delta\theta_i$ and the outputs of the network are decreased to the zero by changing the weights of connections of neurals by "back propagation algorithm".

5-2. Execution Process

After thirty thousand times in learning process, the error obtained by moving the manipulator using the neural network are shown in Table 3. Initial errors are in the upper side of each row and final errors after learning process are in the lower side. Figure 7 shows the simulation results. It is obsedved that the neural network lets the manipulator approach the workpiece. The end-effector can move to the neighborhood of the desired position and orientation.

5-3. Adaptability of neural network

Figure 8 shows the adaptability of the neural network. The current picture is not a learned one, but the neural network lets the manipulator approach the workpiece with desired position and orientation. The neural network is the correct model of the relative function between the end-effector and the workpiece. Several results are shown in Table 4.

6. Conclusion

In this paper a scheme for the control of a robot manipulator with a visual sensor is described. The nonlinear relation between the image data and the control signal for the changes in the joint angles can be learned by an artificial neural network. The validity of this control scheme is confirmed by computer simulations. This approach is efective because it essentially decomposes complex geometric calculations into a simple mapping of the network. Simulations of higher degrees of freedom and implementation of the system on an industrial robot are under planning.

Reference

[1] R.L.Anderson, "Real Time Intelligence Visual Control of a Robot", Proc. of IEEE Int. Workshop on Intelligent Control, pp.89-94, 1985.

[2] A.C.Sanderson, "Vision And Multisensor Feedback Controllers", Proc. of IEEE Int. Workshop on Intelligent Control, pp.80-81, 1985.

[3] W.T.Miller, "Sensor Based Control of robotic Manipulators using a General Learning Algorithm", IEEE J. Robotics Automation, Vol. RA-3, pp.157-165, 1987.

[4] R.Tasi and T.Huang, "Uniquencess and Estimation of Three Dimensional Motion Parameters of Rigid Object with Curved Surfaced", IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-6, pp.13-17, 1984.

[5] J.T.Feddema, "Automatic Selection of Image Features for Visual Servoing of a Robotic Manipulator", Proc. IEEE Int. Conf. on Robottics Automation, pp.832-837, 1989.

[6] Y.Hung, P.Yeh, D.harwood, "Passive Ranging to Known Planar Point Sets", Proc. of IEEE Int. Conf. on Robotics and Automation, pp.80-84, 1985.

[7] D.E.Rumelhart, J.L.McClelland, the PDP Rearch Group, "Parallel Distributed Processing", Bradford Books, Vol. 1 ann 2, MIT Press, 1986.

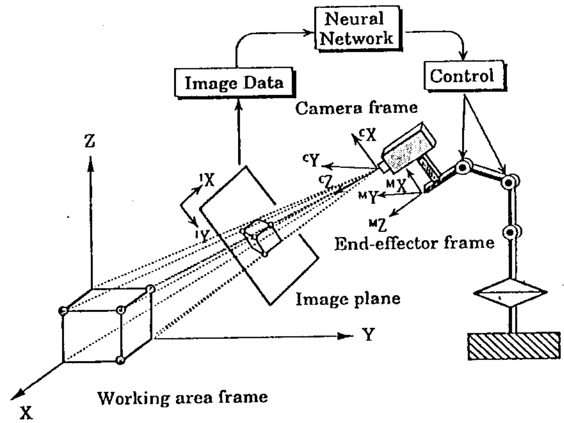

Figure 1 System configuration

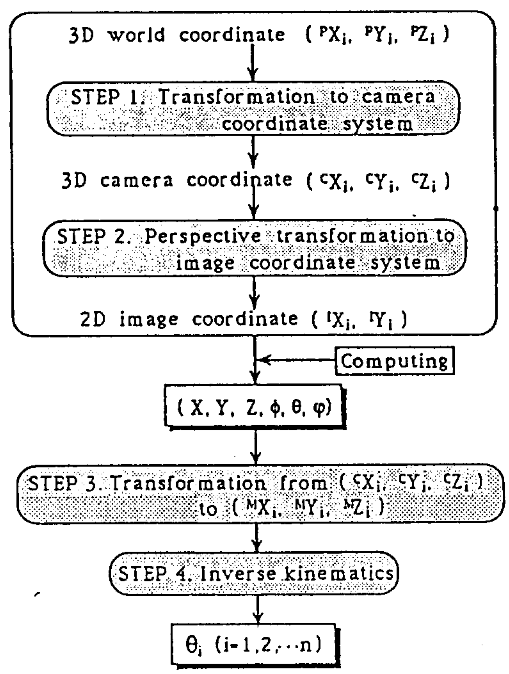

Figure 2 Transformation of coordinate system

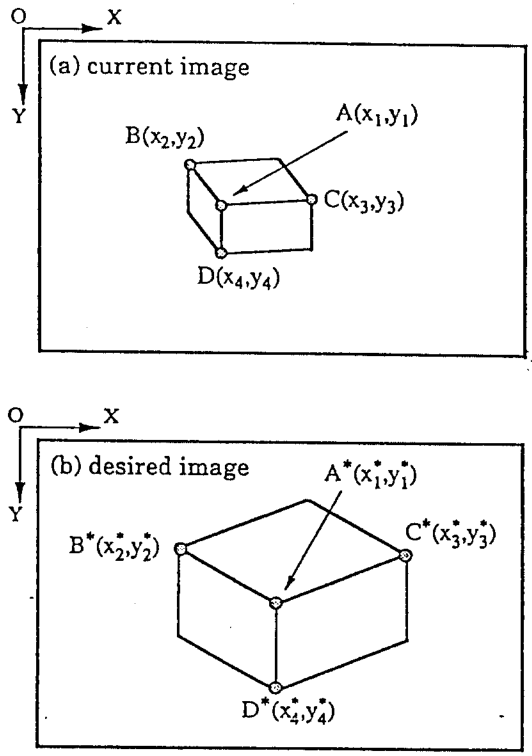

Figure 3 Camera images

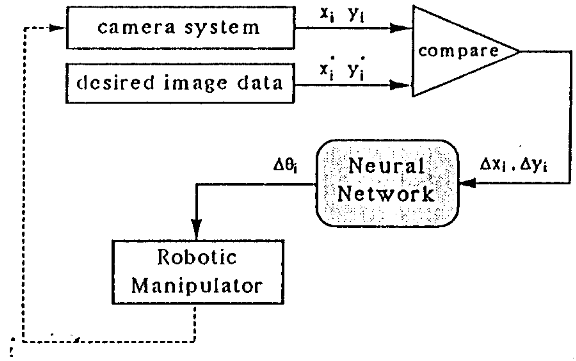

Figure 5 Block diagram of execution process

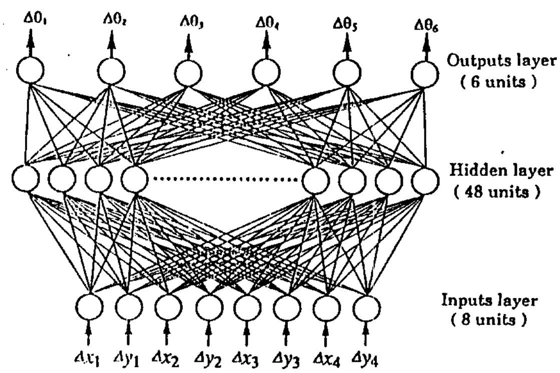

Figure 6 PDP network

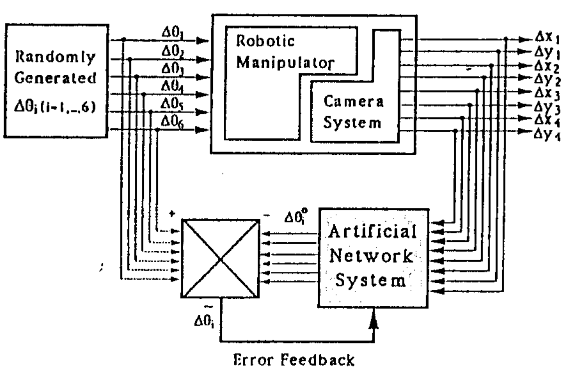

Figure 4 Block diagram of learning process

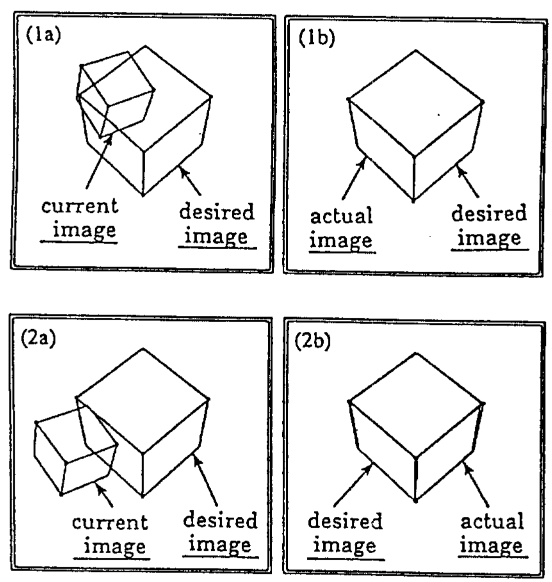

Figure 7 Simulation results (Learned patterns)

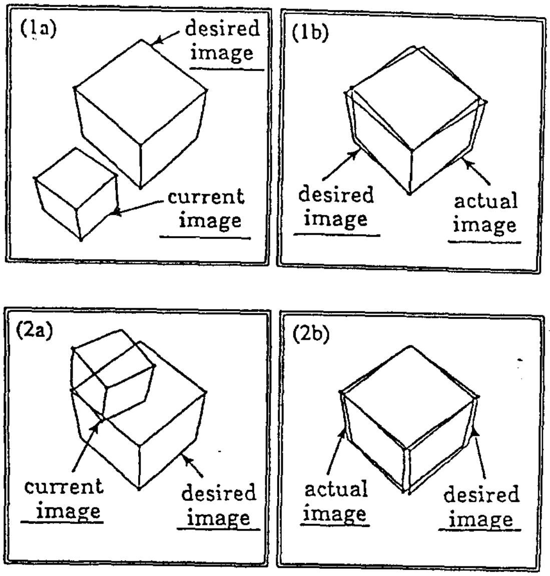

Figure 8 Simulation results (Not learned patterns)

Table 1 Restrictions to training patterns

θ_1	-5.0° ~ 5.0°
θ_2	-30.0° ~ -25.0°
θ_3	20.0° ~ 25.0°
θ_4	-15.0° ~ 15.0°
θ_5	-5.0° ~ 5.0°
θ_6	-15.0° ~ 15.0°

Table 2 Training patterns

Trainig No.	Input Patterns (pixel)								Output Patterns [°]					
	Δx_1	Δy_1	Δx_2	Δy_2	Δx_3	Δy_3	Δx_4	Δy_4	$\Delta\theta_1$	$\Delta\theta_2$	$\Delta\theta_3$	$\Delta\theta_4$	$\Delta\theta_5$	$\Delta\theta_6$
1	-115	6	-108	26	-137	-81	-51	53	-5.0	-29.0	23.0	-13.0	-3.0	-5.0
2	18	-78	42	-55	15	14	108	-38	2.0	-28.0	22.0	-15.0	4.0	-9.0
3	28	-20	29	-2	-14	39	71	40	1.0	-30.0	25.0	-13.0	-2.0	14.0
4	-84	-146	-91	-135	-129	-98	-53	-94	-4.0	-26.0	24.0	-10.0	2.0	9.0
5	-123	-2	-116	16	-141	72	-59	40	-5.0	-26.0	21.0	-8.0	-3.0	-11.0
6	-68	-89	-57	-70	-87	-9	3	-45	-3.0	-30.0	22.0	-7.0	3.0	-13.0
7	-2	-100	-1	-87	-40	-47	39	-49	0.0	-26.0	23.0	-8.0	1.0	13.0
8	24	2	10	23	-48	46	40	80	0.0	-28.0	22.0	4.0	-1.0	13.0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
60	-84	-140	-99	-127	-143	-100	-69	-80	-4.0	-25.0	22.0	1.0	4.0	5.0

Table 3 Image error (Learned patterns)

Error	Δx_1	Δy_1	Δx_2	Δy_2	Δx_3	Δy_3	Δx_4	Δy_4
Initial	-57	-70	3	-45	-87	-9	-68	-89
Final	-1	2	-1	2	0	3	-1	1
Initial	-116	16	-59	40	-141	72	-123	-2
Final	3	0	3	0	4	0	2	1
Initial	8	51	50	86	-31	90	5	33
Final	-2	1	-4	0	-2	0	-2	1
Initial	-1	-87	39	-49	-40	-47	-2	-100
Final	-1	2	-2	1	0	2	-1	2

Table 4 Image error (Not learned patterns)

Error	Δx_1	Δy_1	Δx_2	Δy_2	Δx_3	Δy_3	Δx_4	Δy_4
Initial	-98	101	-56	141	-139	134	-90	80
Final	2	-2	9	-9	9	7	-3	-1
Initial	-49	-91	2	-64	-79	-39	-56	-106
Final	-9	-1	-5	-4	-6	3	-10	-2
Initial	13	-33	48	13	-33	-1	18	-49
Final	-1	-14	-3	-11	-2	-14	0	-11
Initial	52	-15	91	21	13	23	48	-32
Final	-9	5	-12	7	-11	1	-6	3