

Motion Teaching Method for Complex Robot Links Using Motor Current Sensing

Young-bong Bang*, Wonseok Lee*, Kyung-min Lee*, In-su Kim*, Jamie Kyujin Paik*,
Bu-Hyun Shin*

* School of Mechanical and Aerospace Engineering, Seoul National University, Seoul, Korea

(Tel : +82-2-880-1697; E-mail: ybbang@snu.ac.kr)

(Tel : +82-2-880-1726; E-mail: bloodstn@hanmail.net,

genie95@snu.ac.kr, insu03@snu.ac.kr,

jamie79@snu.ac.kr, sbh4462@snu.ac.kr)

Abstract: Robots today have wider application fields than they ever have before. They need to work close to humans and fluid and compliant motions are expected of them. This requires redundant degrees of freedom for completing specific task. And conventional motion teaching method cannot be applied to redundant link structures. In this paper, the authors present a proficient, cost-effective and intuitive method for motion teaching. New software to apply this method to a humanoid is also presented. This new method utilizes current sensors to determine which joints to rotate. The experiment shown in this paper is a case of closed link where arms cannot move independently due to the restrictions in between the hands. After the input of several passing points of motion trajectory, the curve fitting is performed by the developed software. This software can insert new points, delete erroneous points and modify existing points. The developed motion teaching method is applied to the Kumdo robot, which is developed by the authors.

Keywords: motion teaching, Kumdo robot, humanoid, robotic arm, torque sensing, trajectory generation

1. INTRODUCTION

The advancement in technology and its availability to wide range of customers have changed the perception of capability of contemporary robots. With ever increasing advancement in sensors, computer power, and actuators, contemporary robots are expected to serve much diverse tasks than they ever have been. Unlike the conventional industrial robots that perform repetitive, simple and mechanical tasks, more sophisticated moves, pliability, and robustness upon varying environments are needed from robots today. This new generation of robots has created a new category of robots that are able to entertain, help and live with humans. Humanoids fall into this category, and in order for them to mimic humanlike motions, they need extra degree of freedom in their joints. The joints are connected by one or more links to perform specified tasks and moves. If two anthropomorphic arms are controlled, simultaneous motion controls would be in need. With conventional robot teaching methods, it would be impossible to coordinate such moves with joint link structure such as closed link arm. The complexity in the mechanical design in humanoids prevents application of conventional methods. The following are some of the common methods of teaching robots and their problems when they are applied to humanoids.

- Motion capture method:

For this motion teaching method, the desired path and shape of robot's motion is choreographed before-hand by the user. By using markers and/or sensors, the motion paths of pre-marked points are collected and sent to the robot. It is imperative that the human subject who is performing the desired path is of the similar shape and size as the robot itself: also the markings must be placed accordingly. Even for highly priced motion capturing system, the degree of freedom for the acceptable / readable motion range is somewhat limited. And in case of the physical dissimilarity, the motion capturing system becomes very ineffective. Articles [1-2] use varying motion capturing methods to obtain robotic path.

- 6-axis force sensor method:

A 6-axis force sensor can be attached at the end-effector of a robot. The readings of the sensor in each direction are

obtained and this is used to control the rest of the joints in a robot. This sensor is convenient to use when a straight and clean inverse kinematics of certain robot is available. However, when it is the case with a pair jointed arms with many redundant degrees of freedom, this method become ineffective in controlling all the joints.

- Joint torque sensing method:

For this method [3,4], each joint is given a costly torque sensor. Depending on the reading of the sensor, whether it matches the expected values or not, the respective joint can be controlled. However, due to the excessive volume of the torque sensors in each joint, bulky joint design is inevitable. Also, the weight of the links and joint grows with the volume.

The authors of this paper have developed humanoid that can practice Kumdo (Kendo). In spite of the effort to reduce the number of joints in two arms, there is one additional DOF in each arm to make the robot to mimic human moves. A sword is grabbed by two hands, two arm make one closed loop. The teaching method of this paper is developed for the complex robots like this Kumdo robot without additional expensive equipment.

The torque loaded in each joint is measured and compensated for the effect of arm weight. Differently from the existing method, the current sensor attached to the joint motor is used to measure torque. For stable controls, joints only rotate predetermined small angle once the threshold value is crossed. Once the sufficient points are available, the smooth path between these points is generated. For this path generation, software developed in this research is used.

2. KUMDO TRAINING ROBOT

Fig. 1 shows the Kumdo robot developed by the authors of this paper. The Kumdo robot has three DOF in its foot as it is composed of three omni wheels, two DOF in its legs and the arms have four DOF in each. When practicing Kumdo, the direction of the sharp edge of the Kumdo sword (only one of the edges is sharp) determines the orientation of the wrist; therefore, one joint is required to be directly actuated at the left hand while other five joints of two wrists are passively

moved.

Fig. 2 shows the orientation of eight joints of arm. When two arms are crossed with a sword in between, the humanoid's two arms form a closed link. Six DOFs of the sword and the distance between two hands are given, so one active joint in the wrist and six equations for two arms are obtained. Since there are two free DOF left, inverse kinematics alone cannot determine all of the joints' angular positions. Due to these extra DOF, conventional robot motion planning may not be used for this case. Therefore, the authors introduce a method that uses current sensor to control the humanoid. The current sensors can measure the current input to the actuators in each motor and determine the torque load on the joints. This method is firstly applied to motion teaching the arm movements and then to the total system with the sword.

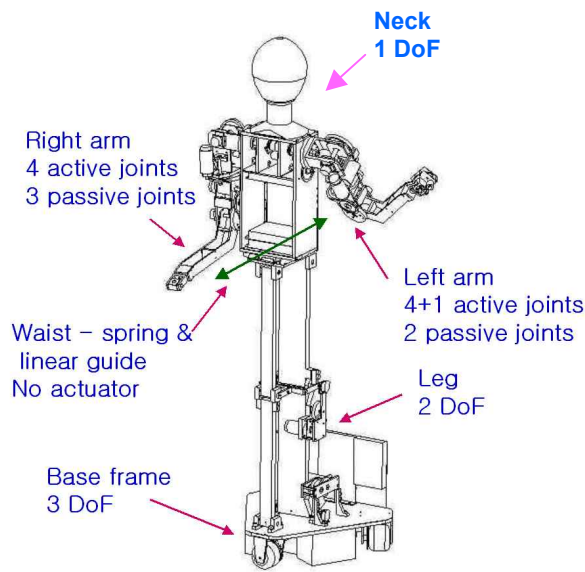


Fig. 1 The developed Kumdo training robot

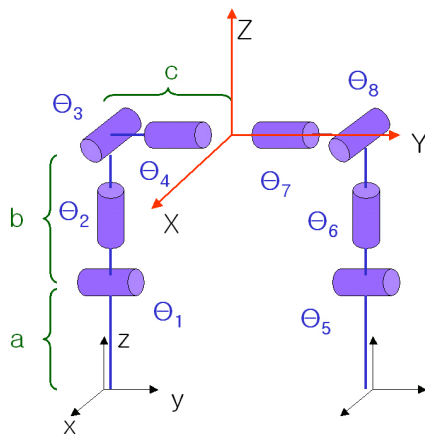


Fig. 2 Joints and links of the developed robot's arms

3. MOTION TEACHING METHOD USING MEASURED ACTUATOR CURRENT

The process of the current detection method in robot and further humanoid applications is shown in the flowchart of Fig.3.

Because the forward kinematics of the robot is already

known, the position of the robot links can be found from each joints' angular orientation measured by the potentiometer attached at each joint. Once the overall position is known, the current requirement of each joint motor to maintain a given position can be calculated. If there is an object to be held in hands, the required current level for the actuators for the directed position will alter as well. After compensating for the effect of gravity at links and objects held by hands, the external forces measured by motor currents are obtained. When the load on the joints reaches the threshold, the motor rotates predetermined angle. The procedure to remove the effect of gravity and additional mass on the links is discussed in the following chapters.

3.1 The torques occurred by the weight of links of an arm

Knowing the angular changes in each joint enables to determine the static torque and reactive forces in each desired position.

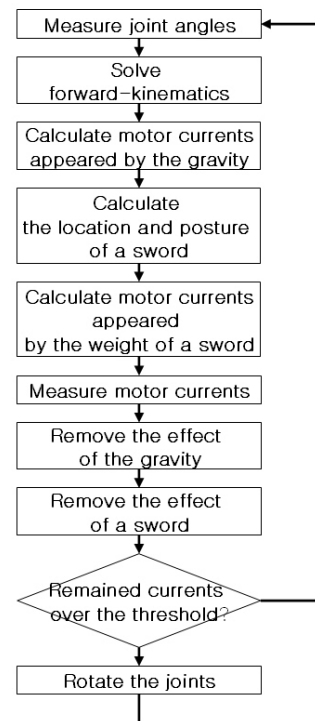


Fig. 3 The flow chart for the motion teaching program

The links and joints are considered to be ideally machined and assembled for simpler calculation. Refining and correcting the erroneous values are done through the several experiments.

Figure 4 shows one of the arms of the robot. Since the joint angles are known, the center of gravity, link/joint coordinates are determined by the forward kinematics.

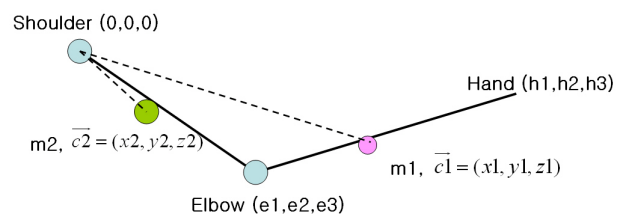


Fig. 4 A model of one arm of the developed robot

The shoulder is set as the origin, and other variables are defined as follows.

- $m1$: mass of link from elbow to hand
- $m2$: mass of link from shoulder to elbow
- $l1$: length of link from elbow to hand (link1)
- $l2$: length of link from shoulder to elbow (link2)
- $\bar{e} = (e1, e2, e3)$: elbow coordinate
- $\bar{c1} = (x1, y1, z1)$: center of mass of mass1(link1)
- $\bar{c2} = (x2, y2, z2)$: center of mass of mass2(link2)

Each coordinate of the defined point changes depending on the position of the arm. Therefore, the new point must be calculated to find the torque.

The center of mass of each link of one arm can be expressed as Eq (1) at the home position.

$$\begin{aligned} \bar{c1}_{\text{home}} &= (0, 0, -l2) + (0, 0, -l1/2) + (p1, q1, r1) \\ \bar{c2}_{\text{home}} &= (0, 0, -l2/2) + (p2, q2, r2) \end{aligned} \quad (1)$$

$(p1, q1, r1)$ and $(p2, q2, r2)$ represent the offset of center of mass (CM) from the geometric center of a link at the home position. Eqs. (2)~(3) represent the transformed CM by the rotational transform.

$$\begin{aligned} \bar{c1}^T &= \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 \\ 0 & 1 & 0 \\ -\sin\theta_4 & 0 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 \\ 0 & \sin\theta_3 & \cos\theta_3 \end{bmatrix} \\ &\cdot \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2)$$

$$\begin{aligned} &\cdot \left(\begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 \\ 0 & 1 & 0 \\ -\sin\theta_4 & 0 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} p1 \\ q1 \\ r1 - l1/2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ l2 \end{bmatrix} \right) \\ \bar{c2}^T &= \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 \\ 0 & 1 & 0 \\ -\sin\theta_4 & 0 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 \\ 0 & \sin\theta_3 & \cos\theta_3 \end{bmatrix} \\ &\cdot \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p2 \\ q2 \\ r2 - l2/2 \end{bmatrix} \end{aligned} \quad (3)$$

The torque that is applied to the elbow by link1's mass is shown in Eq. (4)

$$\begin{aligned} &(\bar{c1} - \bar{e}) \times (0, 0, -m1 \cdot g) \\ &= (x1 - e1, y1 - e2, z1 - e3) \times (0, 0, -m1 \cdot g) \\ &= (-m1g \cdot (y1 - e2), m1g \cdot (x1 - e1), 0) \end{aligned} \quad (4)$$

Torque in the shoulders can be found by adding torques occurred by the link1 and the link2. Eq.(5) and Eq.(6) are the values of torques.

$$\begin{aligned} &\bar{c1} \times (0, 0, -m1 \cdot g) \\ &= (x1, y1, z1) \times (0, 0, -m1 \cdot g) \\ &= (-m1g \cdot y1, m1g \cdot x1, 0) \end{aligned} \quad (5)$$

$$\begin{aligned} &\bar{c2} \times (0, 0, -m2 \cdot g) \\ &= (x2, y2, z2) \times (0, 0, -m2 \cdot g) \\ &= (-m2g \cdot y2, m2g \cdot x2, 0) \end{aligned} \quad (6)$$

The total torque applied to the shoulder due to the gravity alone is $(-m1y1 + m2y2)g, (m1x1 + m2x2)g, 0)$

The torque on each joint is converted to the torque values required from each motor. This can be found from the dot product of the motor axis's direction vector and the torque vector on the joint. The direction vector can be obtained from transforming the initial position as shown in Fig. 2. When the i^{th} joints' rotational axis is \bar{J}_i , each joint's axis of rotation is as found in Eqs. (7)~(10).

$$\bar{J}_1^T = \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 \\ 0 & 1 & 0 \\ -\sin\theta_4 & 0 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 \\ 0 & \sin\theta_3 & \cos\theta_3 \end{bmatrix} \quad (7)$$

$$\begin{aligned} &\cdot \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ \bar{J}_2^T &= \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 \\ 0 & 1 & 0 \\ -\sin\theta_4 & 0 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 \\ 0 & \sin\theta_3 & \cos\theta_3 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned} \quad (8)$$

$$\bar{J}_3^T = \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 \\ 0 & 1 & 0 \\ -\sin\theta_4 & 0 & \cos\theta_4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (9)$$

$$\bar{J}_4 = (0, 1, 0) \quad (10)$$

With the results of Eqs. (7)~(10), the loaded torque on each joint is found as in Eq. (11)~(14).

$$t_1 = \bar{J}_1^T (-m1g \cdot (y1 - e2), m1g \cdot (x1 - e1), 0) \quad (11)$$

$$t_2 = \bar{J}_2^T (-m1y1 + m2y2)g, (m1x1 + m2x2)g, 0) \quad (12)$$

$$t_3 = \bar{J}_3^T (-m1y1 + m2y2)g, (m1x1 + m2x2)g, 0) \quad (13)$$

$$t_4 = \bar{J}_4^T (-m1y1 + m2y2)g, (m1x1 + m2x2)g, 0) \quad (14)$$

The current I in each motor is found by Eq. (15).

$$I = \frac{\text{deceleration efficiency} \times \text{deceleration ratio} \times \text{torque}}{\text{torque constant } (N - m/A)} \quad (15)$$

Therefore, if the measured current does not match what is found by Eq.(15), there is an external load that is applying. The robots links can be rotated according to this measured current level.

3.2 Restriction between two hands

As it was mentioned in previous sections, the two arms of the humanoid often must perform simultaneous tasks and sometimes form a closed link system. In a closed link system, the distance between the hands is restricted to the size of the object it is holding. Also if the object must retain its global orientation, the relative hand positions must have according restrictions. To keep the restriction, only one arm is controlled by the torque input, and the position other arm is calculated by the inverse kinematics.

If the relative position is fixed, the new position of the following hand is directly found by the inverse kinematics. If only the distance between hands is constant, the following hand's new position can be found as shown in Fig. 5. The arm with larger torque input is set to be the principal arm. The following hand's new coordinate is determined as a point on the line between the new principal hand and the old previous hand as well as the distance is set to be the desired size.

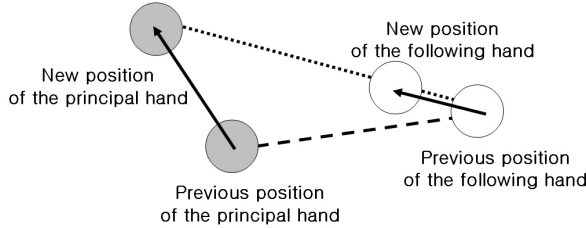


Fig. 5 A new calculated position of following hand

Aforementioned two methods can determine the new position of the following hand. Therefore, if inverse kinematics with additional DOF need to be solved, more parameters are required. In such case, with the measured current levels, undefined rotation of additional joints can be decided. For Kumdo robot, the second joint of each arm takes on this routine.

3.3 Additional mass in arms

When the robot is holding an additional weight other than its own body weight, more things need to be considered. The Kumdo robot holds a sword and the posture and the weight of a sword affect the current levels of joint motors.

Fig. 6 shows the free-body-diagram of the sword held by two hands. As shown in Fig. 6, all directional components of the weight is resisted by the left, while the right hand only holds the radial direction of the force. Both wrists are free to rotate so, there is no reactive torques acting on the wrists.

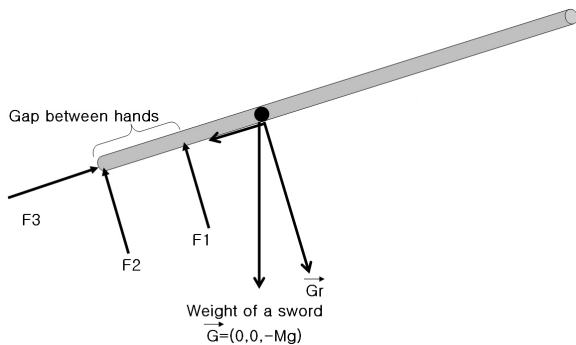


Fig. 6 Free-body-diagram of a sword of the Kumdo robot

Let M be the mass of the sword and L be the distance between the left-hand and the sword's center of mass (CM). When the vector that points the right-hand from the left-hand is \vec{b} and the gravity is \vec{G} , \vec{F}_3 can be found as in Eq. (16).

$$\vec{F}_3 = \frac{\vec{G} \cdot \vec{b}}{|\vec{b}|^2} \cdot \vec{b} \quad (16)$$

To find \vec{F}_1 and \vec{F}_2 , two equilibrium equations for force and the torque are set as Eq. (17)~(18). Here, \vec{G}_r is the sword's

radial component of gravity vector.

$$\vec{F}_1 + \vec{F}_2 + \vec{G}_r = 0 \quad (17)$$

$$\vec{b} \times \vec{F}_1 + \frac{L}{|\vec{b}|} \cdot \vec{b} \times \vec{G}_r = 0 \quad (18)$$

\vec{b} is perpendicular to \vec{F}_1 and \vec{G}_r ; therefore, \vec{F}_1 can be calculated as in Eq. (19).

$$\vec{F}_1 = -\frac{L}{|\vec{b}|} \cdot \vec{G}_r = -\frac{L}{|\vec{b}|} \cdot (\vec{G} + \vec{F}_3) \quad (19)$$

\vec{F}_2 is found using Eq. (20).

$$\begin{aligned} \vec{F}_2 &= -\vec{G}_r - \vec{F}_1 \\ &= -(\vec{G} + \vec{F}_3) \cdot \left(1 + \frac{L}{|\vec{b}|}\right) \end{aligned} \quad (20)$$

Therefore, the total force on right-hand is $\frac{L}{|\vec{b}|} \cdot (\vec{G} + \vec{F}_3)$, and

on the left-hand is $(\vec{G} + \vec{F}_3) \cdot \left(1 + \frac{L}{|\vec{b}|}\right) - \vec{F}_3$.

Consequent calculation is similar to Eq. of chapter 3.1.

4. CURVE FITTING OF THE TAUGHT POINTS

4.1 Curve fitting

In order to smooth out the fitting curve, independently developed software is used. There are largely two methods in fitting curves into discrete points. Firstly the curve is made to pass through all the points by interpolation. And the other is to find the general trend by regression. When interpolation method is used, the created path would need to pass all points and this will make cusps which will in turn distort finesse in the motion. When regression method is used, the starting and the finishing points of motion are likely not to match the initial input points. Therefore, in this research, high-order polynomial regression is used with restrictions of starting and finishing points and velocities.

Eq. (21) shows the generic 9th order polynomial with boundary conditions.

$$y = a_0 + a_1 t + a_2 t^2 + \dots + a_9 t^9 \quad (21)$$

Initial time: $t_1 = 0$

position: y_1

velocity: y'_1

Final time: t_n

position: y_n

velocity: y'_n (n: number of discrete points)

Using the boundary condition, a_0, a_1 are found in Eqs. (22)~(23).

$$f(t_1) = f(0) = a_0 = y_1 \quad (22)$$

$$f(t_1) = f'(0) = a_1 = y'_1 \quad (23)$$

Also, relative Eq (24)~(25) for a_8, a_9 are yielded.

$$a_8 = \frac{1}{t_n^8} (9y_n - t_n y'_n - 9a_0 - \dots - 2a_7 t_n^7) \quad (24)$$

$$a_9 = -\frac{1}{t_n^9} (8y_n - t_n y'_n - 8a_0 - \dots - a_7 t_n^7) \quad (25)$$

When these are substituted in Eq. (21), and reorganized, the error of i th point would be as presented in Eq. (26).

$$e_i = y_i - (a_0 + a_1 t + \dots + a_9 t_i^9) \quad (26)$$

The sum of squares error is shown in Eq. (27)

$$S = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - a_0 - a_1 t - \dots - a_9 t_i^9)^2$$

$$= \sum_{i=1}^n \left\{ y_i - k_{0i} a_0 - \dots - k_{7i} a_7 - (1 - k_{oi}) y_n + \left(\frac{t_i^8}{t_n^8} - \frac{t_i^9}{t_n^9} \right) y'_n \right\}^2 \quad (27)$$

Here,

$$k_{mi} = \left\{ t_i^m - (9-m) \frac{t_i^8}{t_n^{(8-m)}} + (8-m) \frac{t_i^9}{t_n^{(9-m)}} \right\}$$

$$m = 0, 1, 2, \dots, 9$$

For the least-squares fit of the polynomial, we minimize S with respect of the unknown coefficients a_2, a_3, \dots, a_7 . For this, we set the partial derivatives of S with respect to the coefficients equal to zero :

$$\frac{\partial S}{\partial a_2} = -2 \sum_{i=1}^n k_{2i} \left\{ y_i - k_{0i} a_0 - k_{1i} a_1 - \dots - k_{7i} a_7 - (1 - k_{oi}) y_n + \left(\frac{t_i^8}{t_n^8} - \frac{t_i^9}{t_n^9} \right) y'_n \right\} = 0$$

$$\vdots$$

$$\frac{\partial S}{\partial a_7} = -2 \sum_{i=1}^n k_{7i} \left\{ y_i - k_{0i} a_0 - k_{1i} a_1 - \dots - k_{7i} a_7 - (1 - k_{oi}) y_n + \left(\frac{t_i^8}{t_n^8} - \frac{t_i^9}{t_n^9} \right) y'_n \right\} = 0 \quad (28)$$

Eq. (28) can be rewritten as a system of linear equations in the unknowns a_2, a_3, \dots, a_7 as

$$\begin{bmatrix} \sum k_{2i} k_{2i} & \dots & \sum k_{2i} k_{7i} \\ \vdots & \ddots & \vdots \\ \sum k_{7i} k_{2i} & \dots & \sum k_{7i} k_{7i} \end{bmatrix} \begin{bmatrix} a_2 \\ \vdots \\ a_7 \end{bmatrix}$$

$$= \begin{bmatrix} \sum k_{2i} \left\{ y_i - k_{0i} a_0 - (1 - k_{oi}) y_n + \left(\frac{t_i^8}{t_n^8} - \frac{t_i^9}{t_n^9} \right) y'_n \right\} \\ \vdots \\ \sum k_{7i} \left\{ y_i - k_{0i} a_0 - (1 - k_{oi}) y_n + \left(\frac{t_i^8}{t_n^8} - \frac{t_i^9}{t_n^9} \right) y'_n \right\} \end{bmatrix}$$

With Eq. (28), (24)~(25) all coefficients of Eq. (21) is determined.

4.2 Curve fitting and editing program

The software to edit and display the acquired discrete points and fitted curves on screen is also developed in this research. Fig. 7 is the output of a sample robot path and the user interface of the software.

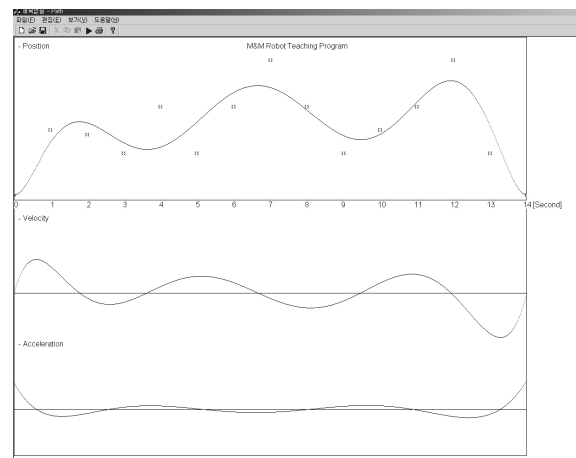


Fig. 7 Curve fit and edit program

This program can produce displacement, velocity and acceleration of each joint as shown in the Fig. 7. In order to improve the quality of the fitted curve, the governing equations can be chosen between 4th, 5th, and 9th order polynomial.

If it is required to edit the input value, it can be selected by a mouse click and moved by a keyboard input. The real time response of the point displacement, velocity and acceleration are displayed on the screen and the user can make appropriate changes to the curve, including deleting and inserting new points. This program was written using Visual C++ and its Microsoft foundation Class (MFC) and has been tested in Windows 2000, 2003 and XP.

5. EXPERIMENTS

5.1 Calibration of Kumdo robot's arm links

The necessary coefficients of the sword and the arms to solve equations in section3 can be directly measured. However, when suitable threshold values is available, and few assumptions are made, the current sensing motion teaching method can be performed without making any external measurement and data preparation.

Firstly, in order to procure mass and the CM of the arm link, the robot is set in four simple positions as shown in Fig.8.

From each simple arm orientations, the input current level is measured and compared to the predicted values which are already available from calculations explained at section 3. When they are analyzed, the coefficients can be approximated.

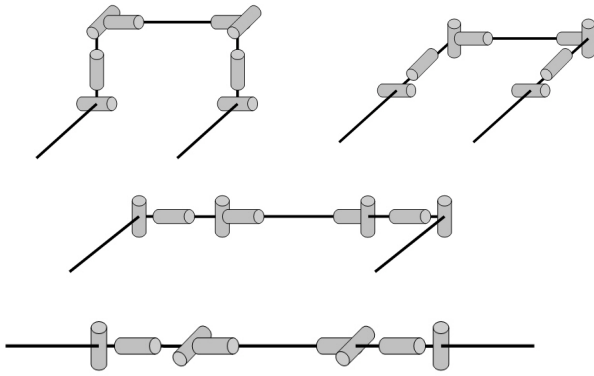


Fig. 8 Four basic posture for variable calibration

The first and the second joint motors require less current than the third and the fourth joint motors. Therefore, the first and the second joints use the sensors that take maximum allowable current input of 2A where it is 4A for the third and the fourth sensors.

5.2 Calibration of a sword held by two hands

Once the coefficients of the links are determined, the weight and the CM of a sword are accordingly found. Eq. (16) shows that the weight of the sword is only resisted by \bar{F}_3 and there is no effect of the CM of a sword when it is positioned perpendicular to the floor. Thus, when the current signal from the left hand is known at the perpendicular position, the sword weight is also found. When a sword is perpendicular to the floor, the CM affects the resist forces. Therefore, measure the torques at two positions, the weight and CM of a sword can be determined. Because this process can be automated with simple motion commands, it is possible that the robot can adapt to the different object for itself.

5.3 Motion teaching

Motion teaching is possible once all the coefficients are



Fig. 9 Teaching of the Kumdo training robot

calibrated. Fig 9 shows the motion teaching process. As mentioned in previous sections, the incremental orientation of joint is fixed. For the experiments, the rotations are made with 0.2 degree increments. The sampling time of all joints is 10 msec, which is enough to make fluid motion in the robot. The torque threshold is 50mNm and this allows a single operator to complete a motion teaching process. Once all the marking points are found through the motion teaching process, motion path is smoothed out using the developed software. This refined motion path is used to produce desired movement in the arms of the Kumdo robot. After the first action, with the developed editor, the motion is perfected to generate more realistic performance.

6. CONCLUSION

In this paper, a new method that uses current detection for motion teaching is presented. Independently developed software allows the user to edit the teaching points and these points are consequently used to generate smooth motion path for the Kumdo robot. Kumdo robot is controlled using this generated motion path.

This new motion teaching method has following advantages compared to the conventional teaching methods

- There is no need for external equipments and other testing apparatus.
- The motion teaching process can be carried out by only one operator regardless of the weight of a robot arm.
- Generally, inverse kinematics of robot links may not be solved. Even when the joint links have complex relationship, this motion teaching method works regardless of the architecture of the robot.
- Current sensors that are used for this method is cheap and easy to install. Also, most industrial amplifiers already detect the input current level; therefore, application of this method is even easier.

It is not only possible to apply this method to the Kumdo robot application, but also for robots that would carry out more complex tasks. Such as having extra restriction in the arm movements and lifting objects are executable with this motion teaching procedure.

REFERENCES

- [1] M. Ringer and J. Lasenby, "Multiple hypothesis tracking for automatic optical motion capture," *Computer Vision -ECCV 2002: 7th European Conf. On Computer Vision*, pp. 524, 2002
- [2] R. Hoshino, D. Arita, S. Yonemoto and R. Taniguchi, "Real-time human motion analysis based on analysis of silhouette contour and color blob," *Proc. of Articulated Motion and Deformable Objects : 2nd Intl. Workshop*, pp.92-103, 2003
- [3] Z. Weng and H. Nishimura, "Final-state control of a two-link car robot," *Advanced Robotics*, Vol.16, No. 4, pp. 325-343, 2002.
- [4] K. Ohishi and H. Nozawa, "Robust motion control with the consideration algorithm of joint torque saturation for a redundant manipulator," *Advanced Robotics*, Vol. 16, No. 4, pp. 345-359, 2002
- [5] N. Dounskaia, C. Ketcham and G. Stelmach, "Commonalities and differences in control of various drawing movements," *Experimental Brain Research*, Vol. 146, No. 1, pp.11-25, 2002