Inverse Kinematics of a Human Arm

Koichi Kondo
Robotics Laboratory Department of Computer Science, Stanford University
Stanford, CA 94305, USA

Abstract

This paper describes a new inverse kinematics algorithm for a human arm. Potential applications of this algorithm include computer-aided design and concurrent engineering from the viewpoint of human factors. For example, it may be used to evaluate a new design in terms of its usability and to automatically generate instruction videos. The inverse kinematics algorithm is based on a sensorimotor transformation model developed in recent neurophysiological experiments. This method can be applied to both static arm postures and human manipulation motions.

1 Introduction

This paper describes a new inverse kinematics algorithm for a human arm. One major application we are pursuing is in computer-aided design (CAD) and concurrent engineering. Most of the research in these areas has focused on geometric modeling, finite element analysis, process planning, and assembly sequencing. We consider a new focus, the human factors component throughout the life cycle of products. Indeed, most products need to be assembled by human workers or customers, and maintained/repaired by service personnel. Computer simulation of human motions will be useful for evaluating the design in terms of its usability and maintainability. Another important application is in producing technical documentations. Almost all products are accompanied by an owner's manual to show how to assemble or operate the product. We believe that rather than a thick cross-referenced owner's manual, explanation is better facilitated through computer animation of the procedure. Simulation of human manipulation motions is a key element for such computer animations.

Consider the problem of automatically generating human arm postures or motions from a description of an object to be manipulated, for example, to evaluate a layout of dials on an operation panel through the simulation of the human arm motions. Human arm postures would need to be computed from given geometry, positions, and orientations of these dials. An appropriate grasp would be selected, and position and orientation of the hand determined. The corresponding arm posture would then be computed. The human arm has seven degrees of freedom even when the shoulder and the torso do not move, and the position and orientation of the hand do not yield a unique solution because of its redundancy. We propose a new approach to realize the inverse kinematics algorithm which automatically selects the natural arm posture. For evaluating ergonomics of a manipulation task, human manipulation motions would need to be simulated. It is widely agreed in the medical community that arm movement is represented kinematically [11], and that dynamics for human manipulation motions are then taken into account as a postprocessing step. We do not need to consider the dynamics unless we need realistic velocity distribution for manipulation motions. However, we do need the inverse kinematics algorithm because most manipulation tasks are specified in terms of the object trajectories.

Our inverse kinematics algorithm is based on a sensorimotor transformation model which describes the process in the brain of how the visual information of the target location (sensor information) is transformed into the arm pointing posture where the tip of the finger touches the target (motor information). This sensorimotor transformation model is derived from experiments on static arm postures in pointing tasks, but we have verified that the inverse kinematics algorithm is applicable to manipulation motions by deriving characteristics observed in curvilinear wrist motions from the sensorimotor transformation model.

In the following sections, we first summarize related work. Second, the inverse kinematics algorithm is described. Third, we derive characteristics of curvilinear wrist motions from the sensorimotor transformation model, and verify that our inverse kinematics algorithm is applicable to human manipulation motions. Finally, we summarize and conclude the paper.

2 Related work

Simulation of human natural movements is related to several different areas of research. We roughly classify related work into two categories, animation of articulated figures and neurophysiology.

Animation is one of the most important topics in computer graphics. In the case of *active* systems such as human figures, a model of motor coordination is crucial for simulating realistic movements. Several gaits have been successfully simulated by using such a model. For example, Bruderlin and Calvert [3] have proposed a hybrid approach to the animation of human locomotion which combines goal-directed and dynamic motion control. McKenna and Zeltzer [9] have successfully simulated the gait of a virtual insect by combining forward dynamic simulation and a biologically-based motion coordination mechanism. However, the motor coordination models for the human arm has not yet been successfully utilized. Some methods have been proposed for the simulation of the human arms and the body. Lee et al. [8] have focused on the simulation of lifting motions, and have proposed a method which considers such factors as comfort level, perceived exertion, and strength. Phillips and Badler [10] have implemented an inverse kinematics technique for interactively controlling the two-footed articulated human figure. Ching and Badler [4] have discussed a motion planning problem for anthropometric figures, but the human-like natural motions have not been obtained for lack of a motor coordination model.

The AnimNL project at the University of Pennsylvania [20] is pursuing a similar goal to ours. The goal of their project is the automatic generation of human figure movements from natural language instructions. Their focus is mainly on planning the sequence of primitive actions from a high-level description of the task, whereas we focus more on the human manipulation motions.

Many researchers in psychology and neurophysiology have been trying to understand how the central nerve system (CNS) solves the coordinated limb motion problem. Soechting [15] gives a good survey of various empirical studies and their results for human arm motions. One important finding in neurophysiological studies is the manner in which the brain determines the arm and wrist posture. The arm posture is determined almost independently of the wrist posture. The arm posture is mainly determined by the object location (the target location), and the wrist posture is then determined by the object orientation and the arm posture. Soechting and Flanders [16, 17] have studied sensorimotor transformations in pointing tasks. They have shown that the visual information about the target is first transformed into some spherical coordinates centered about the shoulder, from which the arm posture is roughly determined. We propose an inverse kinematics algorithm based on this result.

3 Inverse kinematics

This section considers the problem of determining the arm and wrist posture for manipulating an object at a certain position and orientation. We assume that the position and orientation of the torso and the shoulders are given, and that the grasp posture is selected beforehand. We present the algorithm using the right arm for illustration purposes.

Grasp planning is a complicated problem and is closely interrelated to human arm inverse kinematics. However, studies in neurophysiology and psychology suggest that the selection of grasp depend on prior experience and intention [2]. Furthermore, potential grasps are often taken into account when the geometry of objects such as dials are designed. We consider that the selection of grasp is mainly related to the design of objects to be manipulated, and assume that potential grasps are given as an input.

The human arm has seven degrees of freedom, and the position and orientation of the hand do not yield a unique solution because of its redundancy. Furthermore, the position and orientation of the object and the grasp posture do not always specify the position and orientation of the hand completely. For example, a cylindrical object such as a dial can be grasped at an arbitrary orientation around the axis of the cylinder. Our inverse kinematics algorithm selects the natural-looking arm posture even when the position and orientation of the hand is not completely specified.

Neurophysiological studies suggest that information about the location of the object to be grasped specifies the arm posture, and that information about about the size and the orientation of the object dictates hand rotation [1]. This implies that the object to be grasped is first categorized into one of many simple primitive shapes such as cylinder and sphere, which in turn determines the arm and wrist postures. We assume that each geometric primitive is simple enough to be characterized by some information about size and at most one orientation vector. For example, the cylinder is characterized by its radius and the direction of the axis. The center position of the cylinder mainly determines the arm posture, and the wrist posture is adjusted to conform the direction of the axis of the cylinder. In robotics, Tomovic et al. [19] have proposed an approach for synthesizing the control for reaching and grasping objects based on the studies on human motor coordination strategies. They assumed that the target objects are of arbitrary shape and are classified into one of a small number of geometric primitives, and that preshaping and alignment of the hand are determined to conform to the selected geometric primitive shape.

Possible grasp postures can be classified into groups [18] which are called grasp modes. Examples of grasp modes are cylindrical, spherical, and hook grips. We assume that each grasp mode is related to a geometric primitive.

We presume that the arm posture is determined in the following three steps.

1. The object to be manipulated is classified into a geometric primitive, and an appropriate grasp mode is selected. Note that it is enough to approximate a local geometry to be grasped to a primitive, and the object may be represented as a set of primitives.

2. The arm and wrist postures are determined based on the selected geometric primitive and grasp mode.

3. Finger joint angles are adjusted to actually grasp the object.

The first step corresponds to visual recognition, and the last step means fine manipulation based on tactile sensing. This paper does not consider these first and last steps. The inverse kinematics is the second step, and needs information on the selected geometric primitive and grasp mode as an input which is more precisely defined in the following section.

Results from neurophysiology indicate that the arm posture is determined almost independently of the wrist posture. Moreover, the arm posture is determined first, and then the wrist posture is adjusted. Accordingly, it is also known that the arm posture is mainly determined by a simple sensorimotor transformation [16, 17]. Based on these results, we propose a method for solving the inverse kinematics problem in the following way. The arm posture is first estimated by the sensorimotor transformation model, and the wrist joints are activated to conform the hand to the object. Finally, all joint angles are precisely adjusted.

We define wrist joints for the following discussion. Fig. 1 shows three joints. The rotation around the forearm is called forearm spination/pronation. The motion of the second wrist joint is called wrist abduction/adduction. The rotational axis for the wrist abduction/adduction is normal to the palm. Finally, there is wrist extension/flexion whose axis of rotation lies in the palm. The axes for these three rotations intersect each other at the wrist. We define a local coordinate frame of the hand so that the origin of this frame is on this wrist position. We denote this as the *hand frame*.

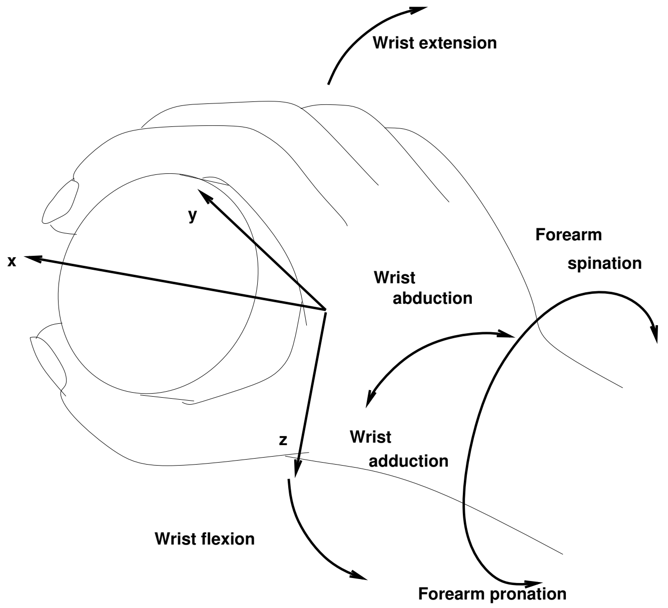

Fig. 1 Wrist joints and cylindrical grip

3.1 Grasp mode

The grasp posture is given as an input to the inverse kinematics algorithm in terms of a relative coordinate transformation between the hand frame and a *grasp frame* which is defined in this section.

The grasp frame is a local coordinate frame which characterizes each grasp mode and shows the relationship between the hand and the geometric primitive. We define the grasp frame so that the x-axis of the grasp frame is related to the orientation vector of the corresponding primitive.

For example, Fig. 1 shows the cylindrical grip and its grasp frame. In this case, the x-axis of the grasp frame is aligned with the axis of cylinder at the grasping position. The y- and z-axes of the grasp frame are less important. Fig. 2 shows another example where the thumb and the index finger pinch a thin rod. This is a grasp mode called pad-to-pad prehension. The grasp frame is defined so that the x-axis is along the pinching direction (Fig. 2). In this case, the x-axis of the grasp frame should be aligned so that it is perpendicular to the axis of the rod, but again the directions of y- and z-axes of the grasp frame are arbitrary. When the primitive, e.g. a sphere, does not have an orientation vector, we define the grasp frame so that the direction of the x-axis shows the typical direction of approach motions.

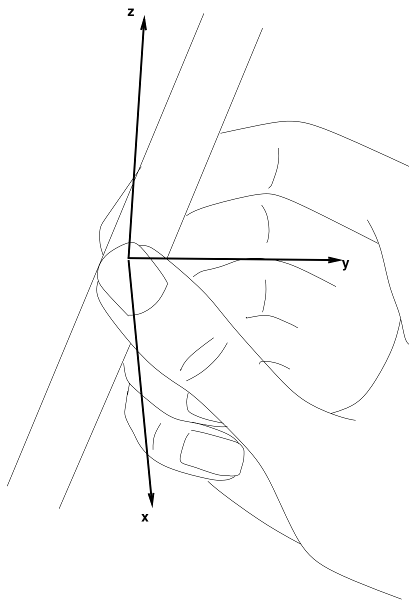

Fig. 2 Pad-to-pad prehension and its grasp frame

The relative coordinate transformation between the hand frame and the grasp frame changes depending on the size of the primitive. For example, larger radius of cylinder results in larger distance between the hand frame and the grasp frame for cylindrical grip. We assume that this coordinate transformation is computed beforehand by considering kinematics of fingers and palm.

3.2 Arm posture

Given a grasp posture, the next step is to estimate the arm posture. Soechting and Flanders [16, 17] have shown that the arm posture is mainly determined by a simple sensorimotor transformation in the brain. Their model suggests that parameters for the arm posture are linearly related to some spherical coordinates centered on the shoulder. Fig. 3 shows these parameters.

Let us denote the coordinate frame centered on the shoulder as a *shoulder frame*. The origin of the shoulder frame is at the shoulder position, the x-axis is along the line that connects the two shoulders, the y-axis is the outward normal of the chest, and the z-axis points towards the head. The parameters for the arm posture are upperarm elevation θ, forearm elevation β, upperarm yaw η, and forearm yaw α.

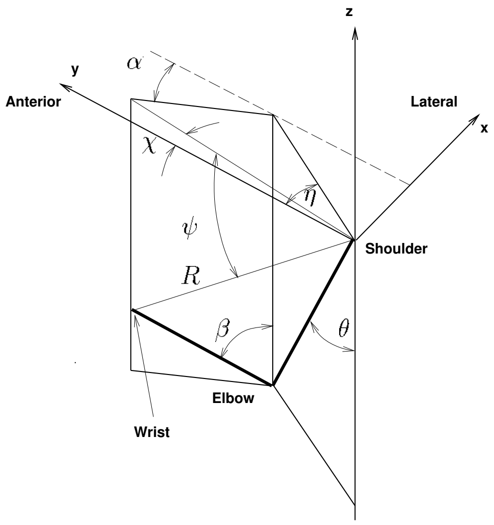

Fig. 3 Parameters for arm posture

The position of the wrist is expressed in terms of the spherical coordinates, azimuth χ, elevation ψ, and radial distance R. The spherical coordinates are related to the coordinates of the shoulder frame in the following way:

$$\left\{\begin{array}{rc} R^2 = & x^2+y^2+z^2 \\ \tan\chi = & x/y \\ \tan\psi = & z/\sqrt{x^2+y^2} \end{array}\right. \tag{1}$$

The sensorimotor transformation model [17] suggests that the parameters for arm posture are approximated by a linear mapping from the spherical coordinates of the hand frame (actually, Soechting and Flanders report this mapping from the position of the end of the stylus held by the test subject, but we simplify this to the hand). The relation is:

$$\left\{\begin{array}{rc} \theta = & -4.0+1.10R+0.90\psi \\ \beta = & 39.4+0.54R-1.06\psi \\ \eta = & 13.2+0.86\chi+0.11\psi \\ \alpha = & -10.0+1.08\chi-0.35\psi \end{array}\right. \tag{2}$$

where the units of measure are centimeters and degrees. Since this is only an approximation, plugging the arm posture parameters back into the forward kinematics of the arm results in a positional error of the hand frame. This is compensated for in the final stage of the inverse kinematics algorithm.

Once the generalized coordinates θ, β, η, and α have been obtained, they are transformed into the four joint angles of the forearm and the upper arm. We check whether they violate any of their limits. If they are within their limits we proceed to find the wrist joint angles. If a limit is violated (an illegal posture), an adjustment phase is performed as described below.

For an illegal posture obtained from Eq. 2, it turns out that the joint angle ξ (the rotation around the upper arm as shown in Fig. 4) is the only one to violate its limits. We correct for this in the following manner. Consider a new set of generalized coordinates of the arm, consisting of the wrist position and the angle ϕ (the rotation of the elbow around the axis between the shoulder and wrist as shown in Fig. 4). By decreasing the value of ϕ from the initial illegal posture (moving the elbow upward), ξ will move back into the feasible joint range without ever changing the wrist position. Note that ξ is obtained by transforming the wrist position and ϕ into the joint angles of the arm.

To summarize, the arm posture is first estimated in the following way:

1. The arm posture is obtained using the transformation of Eq. 2.
2. The values of ϕ and ξ are calculated.
3. If ξ violates the joint limit, then ϕ is decreased until the corresponding ξ satisfies the joint limit.

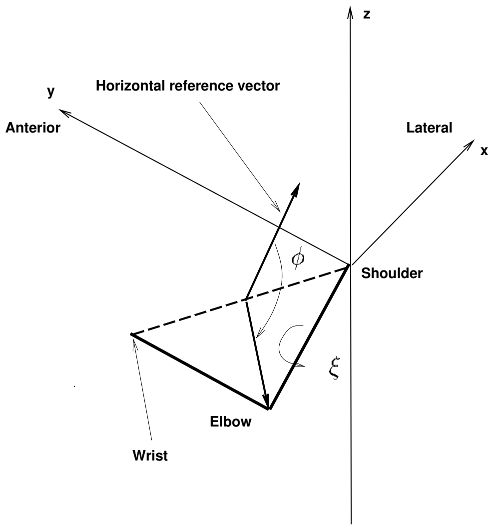

Fig. 4 Elbow rotation and shoulder rotation

3.3 Wrist joints

The next step is to estimate the joint angles at the wrist. In this step, the x-axis of the grasp frame which characterizes the grasp mode is aligned to the axis of the primitive object. We do not consider the directions of the y- and z-axes in this stage. We have found that the natural wrist posture can be simulated by modifying the initial neutral posture with the following strategy. We set the initial wrist orientation to the *functional position*, from which optimal functions are most likely to occur [18]. This position has been established experimentally in the medical community. In the functional position, the wrist is in slight extension (20°) and slight adduction (10°). The joint angle for the rotation around the forearm (forearm spination/pronation) is set to zero (neutral position). When the direction of the x-axis of the grasp frame is arbitrary, the wrist posture is set to be the functional position.

From the casual observation of human arm movements, the desired hand orientation is mainly achieved using wrist joints, with the rotation of the elbow (ϕ) being invoked only when the joints at the wrist are close to their joint limits. We also consider the elbow rotation in estimating the wrist posture. Furthermore, since the range of wrist abduction/adduction is the smallest among the joints at the wrist, and the wrist abduction/adduction is the least likely to be used, we do not move this joint during rough estimation of wrist posture.

Let us consider the example of the cylindrical grasp shown in Fig. 1. In this case, all we have to do is align the x-axis of the grasp frame with the axis of the cylinder. First, the joint angles for forearm spination/pronation and wrist flexion/extension are estimated without considering joint limits so that the desired orientation can be obtained. Two degrees of freedom are sufficient for aligning the x-axis if we allow arbitrary rotation around the x-axis of the grasp frame. In the event that the axis of rotation is close to parallel to the axis to be aligned (x-axis in Fig. 1), forearm spination/pronation or wrist flexion/extension are not changed, because such rotation has a small effect in adjusting the hand orientation. Second, the joint limits are taken into account by stopping the rotation of the joint at its limit. Finally, the elbow rotation is invoked when the desired orientation is not obtained because of the restriction of the joint limits.

Next, let us consider the pad-to-pad prehension shown in Fig. 2. The grasp frame is defined so that the x-axis is along the pinching direction which connects the tip of the thumb to the tip of the index finger. In this case, the only restriction is that the x-axis of the grasp frame should be aligned so that it is perpendicular to the axis of the rod. This means that only one rotational degree of freedom is restricted. We also use the same strategy as the cylindrical grip for this situation. Let us consider the direction of the x-axis of the grasp frame obtained by the arm posture estimated at the first step of the inverse kinematics algorithm and the functional wrist position. We denote this as $\vec{x'}$. The vector $\vec{x'}$ is projected to the plane which is perpendicular to the axis of the rod. We call this vector $\vec{x''}$. The vector $\vec{x''}$ is one feasible direction of the x-axis of the grasp frame at the grasping posture. The wrist joints are adjusted so that the x-axis of the grasp frame is co-linear with $\vec{x''}$.

3.4 Final adjustment

The final step is to adjust all joint angles precisely so that the exact position and orientation of the grasp frame is obtained. Let e be the 6×1 vector which shows the position and orientation error at the end point (the origin of the grasp frame) in the Cartesian space. The 7×1 vector q denotes joint angles. The relation between these two vectors is:

$$e = J(q)\dot{q} \tag{3}$$

where $J(q)$ is the 6×7 Jacobian matrix. This simultaneous equation has only six linear independent relations. Therefore, an infinite number of solutions exist. A solution to this equation can be expressed as follows:

$$\dot{q} = \dot{q}_0 + r\,n \tag{4}$$

where $\dot{q}_0$, r, and n are an instance of $\dot{q}$, a scalar parameter, and the null space of Eq. 3, respectively. In this case, the null space is one-dimensional and is expressed by the multiplication of the scalar r and the base vector n. We employ a constrained optimization to select the $\dot{q}$ which minimizes its norm while satisfying the arm joint limits. The joint angles are then incrementally adjusted accordingly. This ensures that the modification to the hand position and orientation occurs with minimal joint movement. This procedure is applied iteratively until the desired position and orientation of the grasp frame is obtained.

The procedure described above is for aligning the x-axis of the grasp frame with the orientation vector of the primitive. In the cases where the directions of the y- and z-axes are specified, further joint adjustment is necessary to obtain the specified orientation of the grasp frame. This additional step of joint adjustment is performed in the same way as the final step of the procedure.

4 Static posture vs. arm motion

Most manipulation tasks are specified in terms of the object trajectories, and therefore, cannot be simulated without the inverse kinematics algorithm. However, our inverse kinematics algorithm is based on the sensorimotor transformation model which is derived from experiments on static arm postures. In this section, we show that the algorithm is also applicable to human manipulation motions.

It is widely agreed upon that arm movement is represented kinematically [11]. The dynamics or muscle activation patterns are determined in a postprocessing step. Therefore it is not a problem that we do not consider dynamics for human manipulation motions. Lacquaniti et al. [7] have conducted experiments on human handwriting motions, and have shown that the object position is mainly controlled with the shoulder and elbow joints. This suggests that it is reasonable to consider that the object position and orientation are controlled almost independently in human manipulation motions in the same way as our inverse kinematics algorithm. We believe that this justifies the application of our inverse kinematics algorithm to human manipulation motions. Moreover, an analysis of human arm motions shows that the wrist motion is segmented [13] and piece-wise planar [14]. The arm motion for each segment has the same characteristics as a primitive arm motion which results in the elliptic wrist trajectory. This means that the object position is controlled with the same strategy as that for a primitive arm motion which results in the elliptic wrist trajectory.

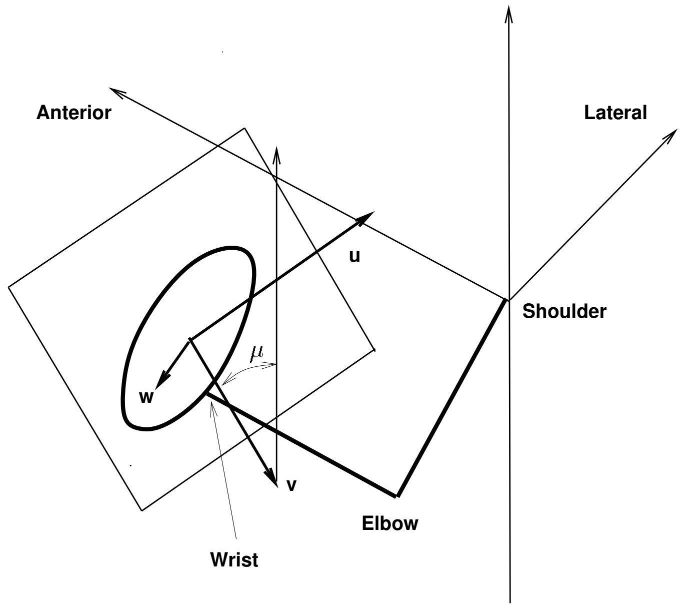

Fig. 5 Elliptic trajectory at wrist

It is then enough to show that the primitive arm motions simulated by our inverse kinematics algorithm match nicely with the experimentally observed ones. We verify this by deriving characteristics of primitive arm motions from the sensorimotor transformation model.

We consider the periodic arm motions which result in elliptic trajectories at the wrist in the following discussion. The elliptic wrist trajectory is on the uv-plane as shown in Fig. 5. The u-axis is horizontal. The normal vector and the origin of the uv-plane are $w = (w_x, w_y, w_z)$ and (x_0, y_0, z_0), respectively. We introduce two parameters μ and ν which are defined by:

$$\begin{cases} \tan\nu = & -w_x/w_y \\ \tan\mu = & -\frac{w_z}{\sqrt{w_x^2+w_y^2}}. \end{cases} \tag{5}$$

The relation between the uv-coordinates and the xyz-coordinates becomes:

$$\begin{cases} x = & x_0 + u\cos\nu + v\sin\mu\sin\nu \\ y = & y_0 + u\sin\nu - v\sin\mu\cos\nu \\ z = & z_0 - v\cos\mu \end{cases} \tag{6}$$

The elliptic wrist trajectory is parametrically defined in the following way:

$$\begin{cases} u(s) = & a\sin s \\ v(s) = & b\sin(s-\sigma) \end{cases} \tag{7}$$

The origin of uv-coordinate frame is the center of the ellipse. If $\sigma = 90°$ and $a = b$, we get a circle. The trajectory will be rectilinear when $\sigma = 0°$ or $180°$.

For describing periodic arm motions, we also use the arm orientation parameters, θ, β, η, and α, shown in Fig. 3. The following invariants have been identified for periodic arm motions in experimental studies [12]:

1. The modulation in these angles (θ, β, η and α) is close to sinusoidal.
2. The phase between the modulation in θ and β is almost equal to $180°$, at least for $0 < \mu < 45°$.
3. The phase between the modulation in α and θ is close to $180° - \sigma$.

Therefore, we can approximate the sinusoidal modulation for θ, β, η and α by:

$$\begin{cases} \theta(t) = & \theta_0 + \theta_1\cos t \\ \eta(t) = & \eta_0 + \eta_1\cos(t-\delta_1) \\ \beta(t) = & \beta_0 + \beta_1\cos(t-\delta_2) \\ \alpha(t) = & \alpha_0 + \alpha_1\cos(t-\delta_3) \end{cases} \tag{8}$$

where $\delta_2 = 180°$ and $\delta_3 = 180° - \sigma$. Since δ_1 is not known, we do not consider $\eta(t)$ in the following discussion.

We will verify that our inverse kinematics algorithm can be applied to human manipulation motions by deriving Eq. 8 from Eq. 7 and the sensorimotor transformation model.

From Eqs. 6 and 7, the elliptic trajectory is parametrically represented in terms of the shoulder frame by:

$$\begin{cases} x = & x_0 + a\cos\nu\sin s + b\sin\mu\sin\nu\sin(s-\sigma) \\ y = & y_0 + a\sin\nu\sin s - b\sin\mu\cos\nu\sin(s-\sigma) \\ z = & z_0 - b\cos\mu\sin(s-\sigma). \end{cases} \tag{9}$$

We then get the following from Eqs. 1 and 9:

$$\left\{\begin{array}{lcl} R^2 & = & x_0^2+y_0^2+z_0^2+(a\sin s)^2+\{b\sin(s-\sigma)\}^2+2a(x_0\cos\nu+y_0\sin\nu)\sin s \\ & & \quad +2b(x_0\sin\nu\sin\mu-y_0\cos\nu\sin\mu-z_0\cos\mu)\sin(s-\sigma) \\ \tan\chi & = & \frac{x_0+a\cos\nu\sin s+b\sin\mu\sin\nu\sin(s-\sigma)}{y_0+a\sin\nu\sin s-b\sin\mu\cos\nu\sin(s-\sigma)} \\ \tan\psi & = & \frac{z_0-b\cos\mu\sin(s-\sigma)}{\sqrt{x^2+y^2}} \\ x^2+y^2 & = & x_0^2+y_0^2+(a\sin s)^2+\{b\sin\mu\sin(s-\sigma)\}^2+2a(x_0\cos\nu+y_0\sin\nu)\sin s \\ & & \quad +2b\sin\mu(x_0\sin\nu-y_0\cos\nu)\sin(s-\sigma). \end{array}\right. \tag{10}$$

Now, we simplify this equation by approximation. We first assume both μ and ν are small (w is close to parallel to the y-axis), and consider $\sin\mu\simeq 0$ and $\sin\nu\simeq 0$. Eq. 10 is then simplified into:

$$\left\{\begin{array}{lcl} R^2 & = & x_0^2+y_0^2+z_0^2+(a\sin s)^2+\{b\sin(s-\sigma)\}^2+2ax_0\cos\nu\sin s-2bz_0\cos\mu\sin(s-\sigma) \\ \tan\chi & = & (x_0+a\cos\nu\sin s)/y_0 \\ \tan\psi & = & \{z_0-b\cos\mu\sin(s-\sigma)\}/\sqrt{x_0^2+y_0^2+(a\sin s)^2+2ax_0\cos\nu\sin s}. \end{array}\right. \tag{11}$$

We further simplify the equation. By assuming that a, b, x_0, and z_0 are small enough relative to y_0 (a relatively small ellipse is located in front of the torso), we regard R as the constant. In the same way, we assume $\sqrt{x_0^2+y_0^2+(a\sin s)^2+2ax_0\cos\nu\sin s}$ is constant. We can also consider that χ and ψ are small enough so that we can assume $\tan\chi\simeq\chi$ and $\tan\psi\simeq\psi$. Then, we have:

$$\left\{\begin{array}{ll} R^2 = & const. \\ \chi = & c_1+c_2\sin s \\ \psi = & c_3-c_4\sin(s-\sigma). \end{array}\right. \tag{12}$$

where c_1, c_2, c_3, and c_4 are constants. Now we apply the sensorimotor transformation model. From Eqs. 2 and 12, the orientation parameters which we are interested in become:

$$\left\{\begin{array}{ll} \theta = & -4.0+1.10R+0.90\,c_3-0.90\,c_4\sin(s-\sigma) \\ \beta = & 39.4+0.54R-1.06\,c_3+1.06\,c_4\sin(s-\sigma) \\ \alpha = & -10.0+1.08\,c_1+1.08\,c_2\sin s-0.35\,c_3+0.35\,c_4\sin(s-\sigma). \end{array}\right. \tag{13}$$

By introducing constants, c_5, c_6, c_7, c_8, c_9, c_{10}, and c_{11}, we obtain:

$$\left\{\begin{array}{lccc} \theta = & c_5-c_6\sin(s-\sigma) & = & c_5+c_6\cos(s-\sigma+\frac{\pi}{2}) \\ \beta = & c_7+c_8\sin(s-\sigma) & = & c_7+c_8\cos(s-\sigma-\frac{\pi}{2}) \\ \alpha = & c_9+c_{10}\sin s+c_{11}\sin(s-\sigma) & = & c_9+c_{10}\cos(s-\frac{\pi}{2})+c_{11}\sin(s-\sigma). \end{array}\right. \tag{14}$$

If c_{11} is small relative to c_{10} so that we can regard $c_{11}\sin(s-\sigma)$ as a constant, we obtain the following:

$$\left\{\begin{array}{ll} \theta(t) = & c_5+c_6\cos t \\ \beta(t) = & c_7+c_8\cos(t-\pi) \\ \alpha(t) = & c_{12}+c_{10}\cos(t+\sigma-\pi) \end{array}\right. \tag{15}$$

where $t=s-\sigma+\pi/2$ and c_{12} is a constant. This means that the phase between the modulation in θ and β is equal to 180°, and that the phase between the modulation in α and θ is to $180°-\sigma$. This result agrees with the experimentally identified invariants for periodic arm motions [12].

We believe that this result justifies the use of the inverse kinematics algorithm for human manipulation motions.

5 Examples

This section shows some of human arm postures computed by this inverse kinematics algorithm. For an animation of human manipulation motions, we refer the reader to [6].

Fig. 6 shows an arm posture for unscrewing the cap from a bottle. The position and orientation of the bottle are given as an input. The rotation around the axis of the bottle of both right and left hands are arbitrary, and the inverse kinematics algorithm determines these hand orientations.

Fig. 7 shows snapshots of an automatically planned human manipulation motion for putting glasses on. The planner has been initially developed for robotic applications [5], but can generate human manipulation motions by using this inverse kinematics algorithm. In this example, the position and orientation of the hand are specified by the planner.

Fig. 8 shows a snapshot of a handwriting motion. Experiments [7] show that the chalk position is controlled with the shoulder and elbow joints, and that the wrist joints are used only for increasing the accuracy when writing large script. Consequently, the wrist joints are fixed throughout the handwriting motion in this example.

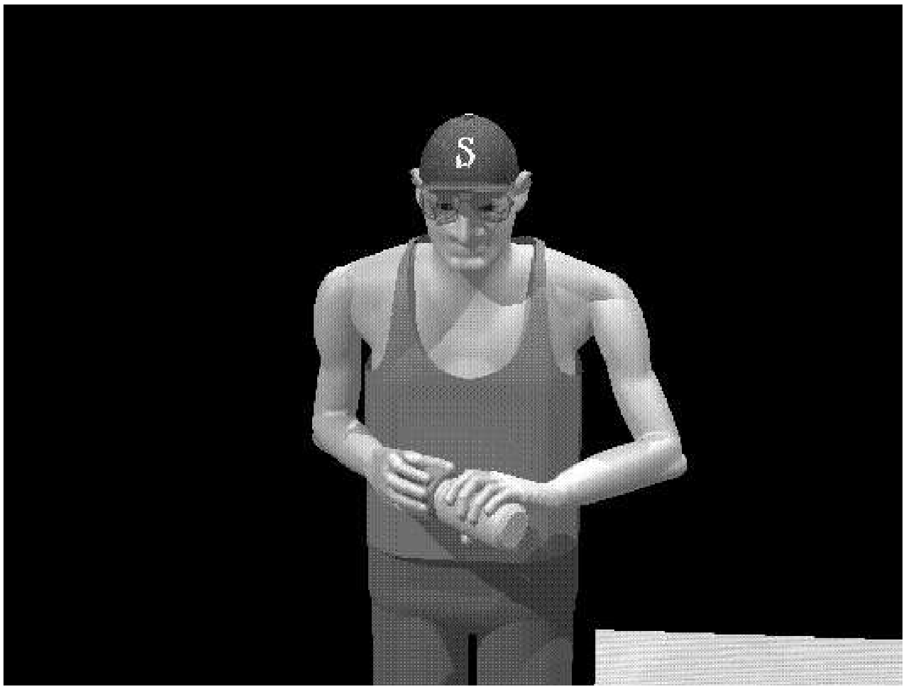

Fig. 6 Unscrewing a cap

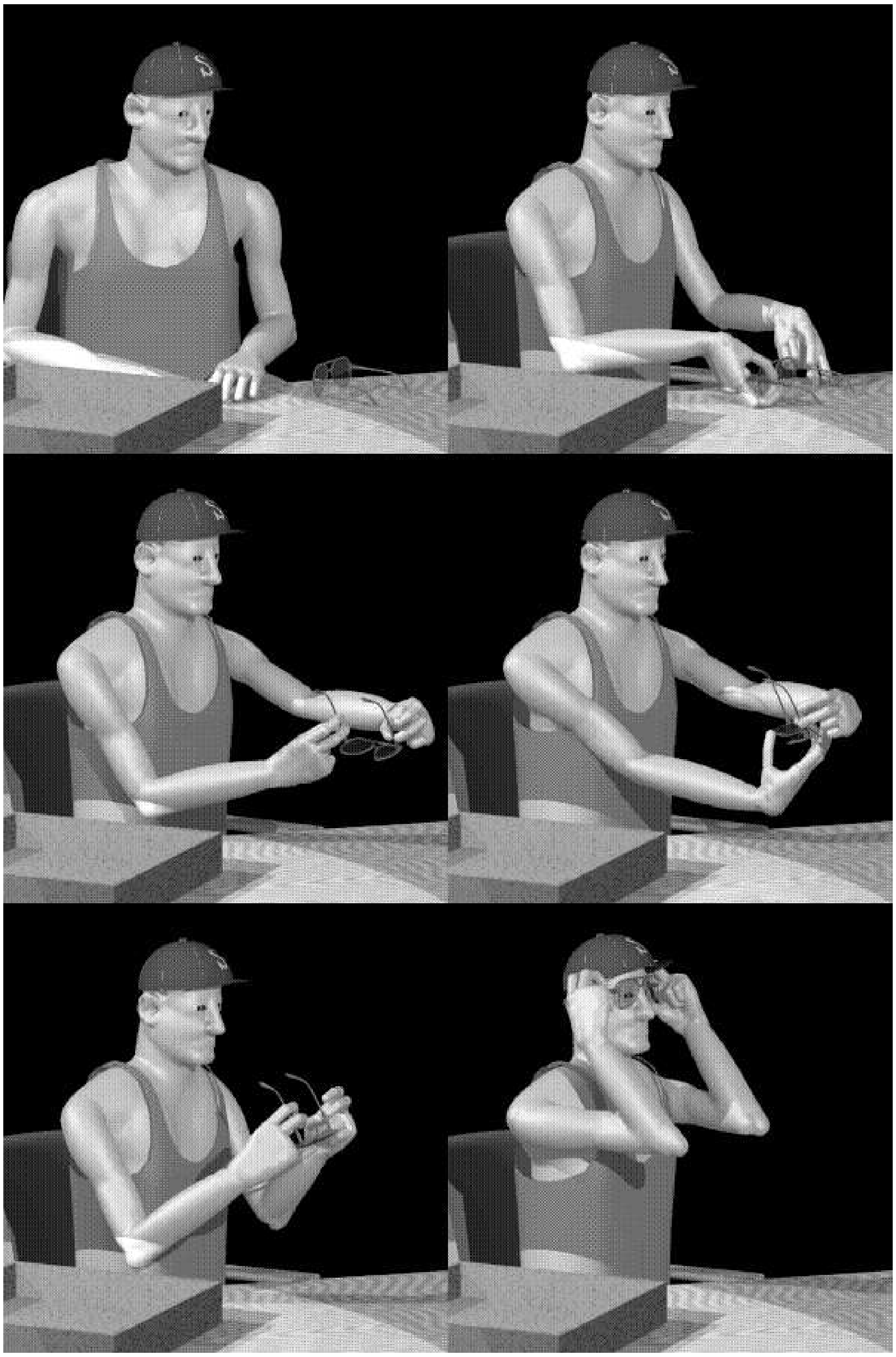

Fig. 7 Manipulating eyeglasses

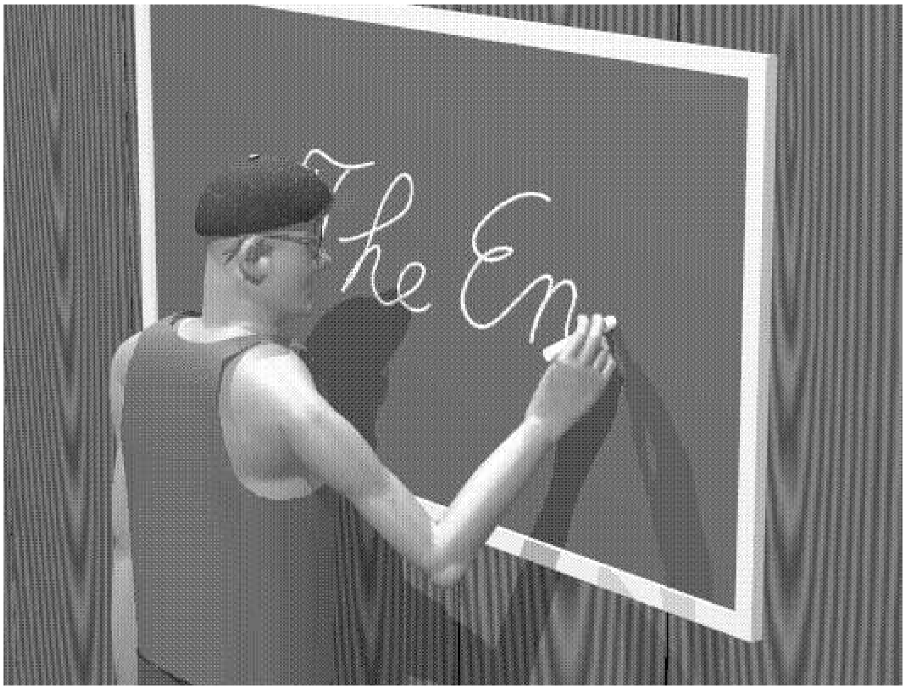

Fig. 8 Handwriting

6 Conclusion

This paper proposes an inverse kinematics algorithm for human arms based on recent experimental results in neurophysiology. The inverse kinematics algorithm takes the position and orientation of the object to be manipulated and information on the grasp mode as an input, and compute human arm posture in the following way. The arm posture is first estimated by the sensorimotor transformation model, and the wrist joints are activated to conform the hand to the object. Finally, all joint angles are precisely adjusted.

This algorithm has been shown to be applicable to human manipulation motions by deriving characteristics of curvilinear wrist motions from the sensorimotor transformation model.

The algorithm is quite reliable and has been successfully integrated with a manipulation planning algorithm [5]. The result is a system that reliably finds the natural human arm manipulation motions.

However, this algorithm is not the *only* method to simulate natural human arm postures and motions. This algorithm may results in collisions between the arm and obstacles in the environment. For example, our algorithm cannot simulate a task where the arms must place an object deep into a tight box. We need another inverse kinematics algorithm that utilize the redundant degree of freedom for the purpose of avoiding obstacles.

In the future, we hope to explore and devise other inverse kinematics algorithms for the arms, and to construct a system that simulates a wide variety of human motions as a key component of concurrent engineering environment.

References

[1] M. Arbib, "Perceptual structures and distributed motor control," *Handbook of Physiology,* edited by V. B. Brooks, Sec. 1. The nervous system: Vol. 2. Motor control, 1981, pp.1449-1480.

[2] S. Athènes and A.M. Wing, "Knowledge-directed coordination in reaching for objects in the environment," *Perspectives on the Coordination of Movement,* edited by S.A. Wallace, Elsevier Science Publishers, Amsterdam, 1989, pp.285-301.

[3] A. Bruderlin and T.W. Calvert, "Goal-directed, dynamic animation of human walking," *Computer Graphics,* Vol. 23, No. 3, 1989, pp.233-242.

[4] W. Ching and N. Badler, "Fast motion planning for anthropometric figures with many degrees of freedom," *Proc. 1992 IEEE Int. Conf. on Robotics and Automation,* 1992, pp.2340-2345.

[5] Y. Koga and J.C. Latombe, "On multi-arm manipulation planning,", *1994 IEEE Int. Conf. on Robotics and Automation.*

[6] K. Kondo, Y. Koga and J.C. Latombe, "On manipulation planning and natural human arm motion,", *1994 IEEE Int. Conf. on Robotics and Automation, Video Proceedings.*

[7] F. Lacquaniti, G. Ferrigno, A. Pedotti, J.F. Soechting, and C. Terzuolo, "Changes in spatial scale in drawing and handwriting: kinematic contributions by proximal and distal joints," *J. Neuroscience,* Vol. 7, No. 3, 1987, pp.819-828.

[8] P. Lee, S. Wei, J. Zhao, and N.I. Badler, "Strength guided motion," *Computer Graphics,* Vol. 24, No. 4, 1990, pp.253-262.

[9] M. McKenna and D. Zeltzer, "Dynamic simulation of autonomous legged locomotion," *Computer Graphics,* Vol. 24, No. 4, 1990, pp.29-38.

[10] C.B. Phillips and N.I. Badler, "Interactive behaviors for bipedal articulated figures," *Computer Graphics,* Vol. 25, No. 4, 1991, pp.359-362.

[11] A.M. Smith et al., "Group Report: What Do Studies of Specific Motor Acts Such as Reaching and Grasping Tell Us about the General Principles of Goal-Directed Motor Behaviour?'," *Motor Control: Concepts and Issues,* D.R Humphrey and H,J, Freund, eds., John Wiley and Sons, New York, 1991, pp. 357-381.

[12] J.F. Soechting and C.A. Terzuolo, "An algorithm for the generation of curvilinear wrist motion in an arbitrary plane in three-dimensional space," *Neuroscience,* Vol. 19, No. 4, 1986, pp.1393-1405.

[13] J.F. Soechting and C.A. Terzuolo, "Organization of arm movements. motion is segmented," *Neuroscience,* Vol. 23, No. 1, 1987, pp.39-51.

[14] J.F. Soechting and C.A. Terzuolo, "Organization of arm movements in three-dimensional space. wrist motion is piecewise planar," *Neuroscience,* Vol. 23, No. 1, 1987, pp.53-61.

[15] J.F. Soechting, "Elements of coordinated arm movements in three-dimensional space," *Perspectives on the Coordination of Movement,* edited by S.A. Wallace, Elsevier Science Publishers, Amsterdam, 1989, pp.47-83.

[16] J.F. Soechting and M. Flanders, "Sensorimotor representations for pointing to targets in three dimensional space," *Journal of Neurophysiology,* Vol. 62, No. 2, 1989, pp.582-594.

[17] J.F. Soechting and M. Flanders, "Errors in pointing are due to approximations in sensorimotor transformations," *Journal of Neurophysiology,* Vol. 62, No. 2, 1989, pp.595-608.

[18] C.C. Norkin and P.K. Levangie, *Joint Structure & Function: A Comprehensive Analysis,* F.A.Davis Company, Philadelphia, 1983.

[19] R. Tomovic, G.A. Bekey and W.J. Karplus, "A strategy for grasp synthesis with multifingered robot hands," *Proc. 1987 IEEE Int. Conf. on Robotics and Automation,* 1987, pp.83-89.

[20] B. Webber et al., "Doing what you're told: following task instructions in changing, but hospitable environments," *SIGGRAPH '93 Course note 80 "Recent Techniques in Human Modeling, Animation and Rendering"*, 1993, pp. 4.3-4.31.