Multisensory learning cues using analytical collision detection between a needle and a tube

F Wang[1], T Poston[3], C L Teo[1], K M Lim[1,2] and E Burdet[1,2]

[1]Dept of Mechanical Engineering and [2]Division of Bioengineering, National University of Singapore

[3]Johns Hopkins Singapore

Email:g0202012@nus.edu.sg (Wang Fei), e.burdet@ieee.org   http://guppy.mpe.nus.edu.sg/∼eburdet/

Abstract— **We are developing a Virtual Reality based training system for micromanipulation in collaboration with the National University Hospital in Singapore. While conventional approaches judge a virtual environment by its resemblance to the real environment, we use simple environments with only selected features of the real task, and develop fast algorithms to investigate the learning of dexterity primitives using various multi-sensory cues. For the needle maneuvering primitive, this paper introduces a method using stereographic projection to compute the distance between the curved needle and a curved tube, necessary to investigate multi-sensory cues systematically. This analytical algorithm is shown to be faster by orders of magnitude than numerical ones. The computation time barely increases with increasing precision, a critical condition to simulate the microworld.**

I. INTRODUCTION

As autonomous robots can perform only highly stereotyped manipulations, most manipulation tasks must be carried out by humans. Micromanipulation is critical to many growing technologies, such as Microsurgery, Micro-electronics, MEMS, Life Sciences, and Nanotechnology, with corresponding needs for skilled manpower. Humans have excellent inference and sensorimotor capabilities, are very versatile and cost less than highly complex robotic systems. However, a human requires significant learning to manipulate objects skillfully under a microscope, for various reasons:

- The operator neither feels nor hears a tool interact with its target, and must rely entirely on visual input and proprioception to control motor activity. The force involved is generally very small, so that haptic feedback is limited.
- The micro world requires greatly modifying visuomotor coordination. The image from the microscope is not only larger than the hands' environment, but also differently oriented, and has less depth of field than normal vision.
- The operator must operate indirectly, not with fingers but via tools such as forceps, scalpel, laser, and micropipette.
- For accurate control, the trainee must develop new motor strategies involving only the fine motion of the fingers. Arm movements induce too much tremor.

Currently trainees typically learn micromanipulation directly on the task, or with real material. For example, microsurgery is learned on latex sheet and then on rats (Fig. 1, left). One must simultaneously control tremor, find a new position for the arms, adapt to the microscope image, recognize the rat's anatomy, *etc*. If the learning is inefficient it is hard for

Fig. 1. Microsurgery learning of rats (left) may be reduced by virtual training on a workstation (right).

the instructor to identify problems; for safety and cost reasons, one cannot try all possible strategies.

To help training manipulation under optical microscope, we are developing a Virtual Reality based training system (Fig. 1, right) as a tool for hand microsurgeons at the National University Hospital. We focus on suturing such as blood vessel anastomosis, performed under 5 to 20 times magnification. Although haptic feedback is negligible in real surgery, we are investigating its potential as a hyper-real cue in learning. The use of haptics implies a need for haptic speeds, so every part of the application must be tuned to take the least time possible.

This paper first describes the learning approach we use, based on representative tasks decomposed into simple dexterity primitives learned using various multisensory cues. It then details the implementation of one such dexterity primitive: maneuvering a needle along a curved path, represented in our learning environment by a curved tube the needle should stay inside without touching. This is not a surgical subtask in itself, just as repeatedly typing `asdf` is not a typist's function. Exercises are to train subskills. We present a very fast algorithm to measure the distance and detect collisions between a needle and a vessel. This algorithm and the program to learn this dexterity skill will be demonstrated at the conference [xxxWF: Still not sure if it could be done]. Another critical task of microsurgery, knot tying, is described in [1].
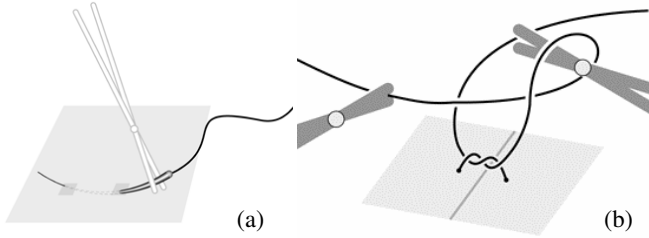
Fig. 2. Two typical tasks of microsurgery: maneuvering a needle accurately with a remote grip (a), and knot tying (b).

II. LEARNING COMPLEX TASKS BY TRAINING SEPARATE DEXTERITY SKILLS WITH MULTISENSORY CUES

A. Multisensory learning cues

Many Virtual Reality trainers aim at realistic emulation, *e.g.,* laparoscopic surgery trainers that display the current position of the tool and shape of the organ so that the apprentice surgeon can operate virtually and feel somewhat realistic interaction forces. The virtual environment corresponding to real micromanipulation would be purely visual [9], [10], as the interaction forces are generally negligible with no 'natural' haptic or audio component. In virtual training, however, we can create additional haptic or audio cues to help acquiring a skill. These cues may be pseudo-realistic (the sound or the force at a multi-point impact when the tool hits a narrow tube around the intended path), or iconic (highlighting, the click of a Geiger counter, etc.) [18], [23].

As the success of a manipulation task depends on neuromechanical control of the arm during the whole motion, we propose using haptic cues in the form of movement conditions. In particular, we have recently shown that motion and path guidance can facilitate learning hand motion for Chinese handwriting [3] and for object handling [5]. We have also discovered [4] that humans respond to unstable dynamics with neural strategies for improved accuracy (Fig. 4a). We will test how these haptic cues, as well as amplified tremor, can help micromanipulation learning. Audio signals related to trajectory error relative to a desired movement will be used as audio cues.

The trainee learning a task acquires a memory or internal model [15] of this task and the conditions under which it must be performed. This internal model persists [6] even in the absence of learning cues, enabling skill transfer to a real task. This results in feedforward control, free from the long delay of visual control [19], [20], similar to the mechanisms we studied in [4]. Neuroscience studies suggest that learning this internal model requires compensating for the delay of the motor-sensory loop [16]. As the visual loop has a particularly long delay of 200*ms* or more [24], we will investigate predictive visual cues that compensate for this delay. In one such case, we will extrapolate the error at the end of a movement from the past and present states and signal it to the trainee, allowing early correction. Could prediction made apparent by a visual or acoustic signal be as efficient as haptic cues? As haptic feedback is relatively costly to provide, such questions are

Fig. 3. Sliding an arc along a circular path is one of the dexterity primitives we use to learn maneuvering a needle in microsurgery

critical to the design of micromanipulation trainers.

B. Dexterity primitives

We have studied typical tasks from microassembly [7] and microsurgery and have selected representative micromanipulation tasks, which require correct perception of position, orientation and also of topology (inside, outside, loop, *etc.*) in 3D, and have both motor and cognitive aspects. Fig. 2 shows two of these tasks typical of microsurgery. In suture needle insertion a curved needle, suturing together two membrane edges, must pass through at carefully planned places. Knot tying is a bi-manual task requiring skillful control of the nylon and knowledge of the correct topology.

From these complex tasks we have extracted simple dexterity primitives corresponding to basic movements they are composed of. Studies suggest [14] that humans may form internal models of these primitives which they can combine in more complex tasks, so that when this task is decomposed and learned in simple steps than all at once, learning a complex task can be faster and performance better. This confirms practice in (for instance) typing courses, which address subtasks of keyboard use before whole words are typed. We in particular want to train motorics separately from cognition, and simple motions one at a time. We will study the effect of multisensory cues on the learning of these primitives and later on their
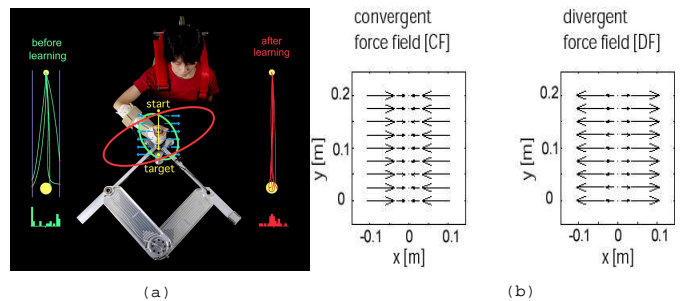

Fig. 4. By training in unstable dynamics, we have found that humans adapt the feedforward motor command to decrease motion deviation [4]. Similarly, we will train micromanipulation dexterity using a divergent force field along the movement. Another haptic cue will be a convergent field guiding motion.

combinations, finally testing the whole learning process on the selected representative tasks.

C. Maneuvering an arc along a tube

In the following we focus on the learning of needle maneuvering, a task requiring very fine control. Both position and orientation must change simultaneously, in six-degree-of-freedom motion. One dexterity skill which will be used to train this task consists of moving a curved needle precisely along its own form, so that as it passes through tissue at two points (Fig. 2a) it does not waver at these points and tear it. This subskill is well represented by the non-surgical task of moving a curved needle along a curved tube of the same radius, without touching the wall (Fig. 3). This presents a more complete test of the required motor skill than does a two-point penetration. By testing the motion all along the length of the needle, it adds situational hyper-reality to the learning of the subtask.

This will be trained with multisensory cues. For example, if the curved needle is going to touch the tube, an audio warning cue may help the trainee correct the motion trajectory, with volume or pitch rising as distance to the tube decreases. (Time discrimination in hearing is acute, so that like haptics this requires fast computation.) A needle that touches the virtual tube causes an error signal. Other cues that will be investigated include:

- visual cues: display of the path, or limits to motion, display of deviation;
- audio cues: contact with virtual objects; error;
- haptic cues: motion guidance, path guidance, destabilization from the intended path.

The implementation of all these cues requires computation of the distance between the needle, modelled as an arc circle, and the vessel, modelled as a portion of a torus. This computation must be very fast, in order to save CPU cycles for the implementation of the cues.

III. FAST ALGORITHM FOR TORUS-ARC DISTANCE

A. Methods to compute intersections

The problem of intersection computation between surfaces and curves is fundamental in computer graphics and geometric modelling. For low degree algebraic surfaces such as quadrics, a number of applications and specialized algorithms have been developed for intersection computations by analytic methods [2], [12], [17], [21]. An arc-shaped needle touching a plane is simple. More complex is touching or approaching a curved surface.

If both needle and surface have complex shapes, the objects can be decomposed into triangles and the distance can be computed from all the distances between the triangles approximating these two objects (Fig. 5a). If the surface is describable by an equation $f(x,y,z) = c$ for a rapidly computable f such as a polynomial, a trivial algorithm evaluates $f(\cdot) - c$ for regularly spaced points on the needle (Fig. 5b) and tests for proximity to 0 (or for a change of sign, which implies crossing 0). Finally, if both surface and needle have tractable shapes,
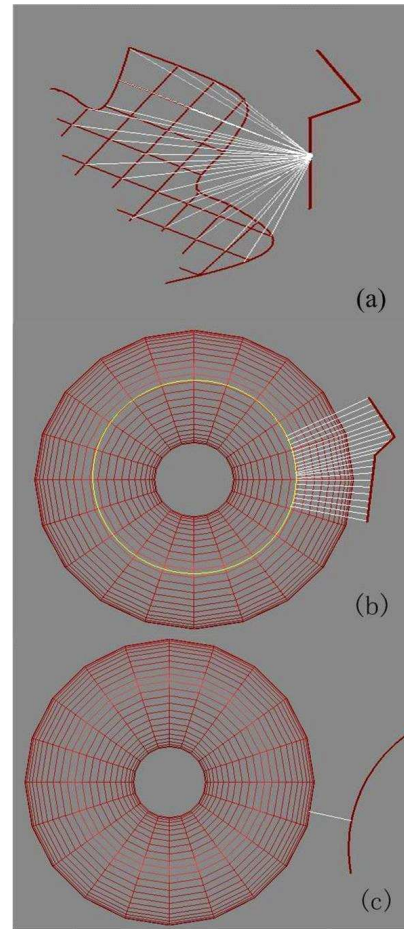

Fig. 5. Three types of method to detect collision between a surface and a needle. (a) distance computation to many surface points for each needle point; (b) For an implicitly defined surface, the iteration can be reduced to the one-dimensional needle; (c) For algebraically simple surface and needle, the distance can be computed analytically.

the relative distance can be computed analytically (Fig. 5c). This approach is the least general, depending on the specific shapes involved, but (analogously to assembly programming for critical tasks) allows the highest speeds. This section develops such a method, and compares the three approaches. By convention, the following developments use bold vector **v** and bold capital matrices **M**.

B. Principle of the algorithm

To train in maneuvering an arc needle along the circle C it lies on, we surround the radius-R circle in the (x,y)-plane by a piece of regular torus. (This will be rotated into various world coordinates, varying the muscular patterns involved in the required hand motion.) There are various ways to describe this by an implicit equation. A torus uniformly r from the circle is given by the fourth degree equation

$$g(x,y,z) = \left(x^2+y^2+R^2+z^2-r^2\right)^2 - 4\left(x^2+y^2\right)R^2 = 0 \tag{1}$$

which leads to unhandy algebra: quartic equations do have closed form solutions, but they are rarely used and involve

roots of various orders. To make the distance analytically tractable, we use a stereographic projection and lift this torus to S^3, where it becomes a cone described by a single second order equation. Solving a quadratic is a high school problem, and requires only square roots, nowadays hardware optimized to a single clock step.

Stereographic projection (*SP*) is a correspondence between spherical and flat geometry, known to geographers since ancient Greek times and widely used in mathematics: a point on a plane tangent to the sphere at its South Pole corresponds to a point on the sphere if the line passes through the North Pole. It has the useful property that circles in the sphere project to circles in the flat domain, and *vice versa* (except that circles through the North Pole correspond to straight lines, 'circles through ∞') [11]. The same property follows for the projection P one dimension higher, between the sphere S^3 of quadruples (l,m,p,q) with $l^2+m^2+p^2+q^2=1$ to flat coordinates (x,y,z) in flat three-dimensional space $\mathbb{R}^3$ — where our tube exists — though the centers of the spherical circles do *not* map to the centers of the circles in $\mathbb{R}^3$. Setting $\rho^2 \equiv x^2+y^2+z^2$,

$$P(l,m,p,q) = \left(\frac{l}{1-q}, \frac{m}{1-q}, \frac{p}{1-q} \right) \tag{2}$$

$$P^{-1}(x,y,z) = \left(\frac{2x}{\rho^2+1}, \frac{2y}{\rho^2+1}, \frac{2z}{\rho^2+1}, \frac{\rho^2-1}{\rho^2+1} \right). \tag{3}$$

Define a toroidal tube in S^3 by the quadratic equation

$$\alpha^2 \left(l^2 + m^2 \right) - \left(p^2 + q^2 \right) = 0 \tag{4}$$

Setting $\alpha = 0$ giving the degenerate case of the unit circle in the (l,m)-plane, which projects to the unit circle U in the (x,y)-plane: $\alpha > 0$ thickens it. (See Appendix for details.) Tori and their circular cross-sections go by P to tori and to circles around (but not centred) on U. They are best parametrized in §3 coordinates, setting

$$(l,m,p,q) = (\cos\phi, \sin\phi, \alpha\cos\psi, \alpha\sin\psi). \tag{5}$$

For drawing, a torus or piece of torus is easily triangulated in these coordinates, and the mesh projected by P to $\mathbb{R}^3$.

To visualize this, consider the analogues in lower dimensions. Fig. 6a shows stereographic projection from the circle S^1 to the line $\mathbb{R}^1$. The restriction $l^2 - q^2$ of the quadratic (4) with $\alpha = 1$ divides the circle into positive regions (blue) and negative (red), vanishing on four boundary points, corresponding to points (not where $l^2 - q^2 = 0$) also marked green in $\mathbb{R}^1$. Moving to projection from the sphere S^2 to the line $\mathbb{R}^2$ (Fig. 6b) spins the first figure, with pairs of points where $l^2 - q^2 = 0$ becoming circles where $l^2 + m^2 - q^2 = 0$. Going to (4) spins the picture in the (p,q) plane, so that the pair of circles becomes one torus: each pair of points with the same (l,m) and opposite q becomes a (p,q) circle with fixed (l,m).

A typical circle (brown in Fig. 6b) in the plane $\mathbb{R}^2$ crosses one of the green circles if and only if the corresponding circle meets $l^2 + m^2 - q^2 = 0$, which is a quadratic calculation also usable for an arc, modelling a needle in the plane. Similar principles apply in $\mathbb{R}^3$, which enables computation of the distance to the needle using the following steps:

- Lift the torus $g = 0$ to a set T in the unit sphere $S^3 = \{x \in \mathbb{R}^4, |x| = 1\}$ where it is the intersection of S^3 with a quadratic cone $G = 0$, around the unit (l,m)-plane circle, as was described above.
- The circle C lifts to a circle $\widehat{C}$ in S^3. Work in the plane π containing $\widehat{C}$ (§III-C).
- The torus equation restricts to an inhomogeneous quadratic on π, which vanishes on $\widehat{C}$ where $\widehat{C}$ meets T (§III-D).

C. Lift the needle to S^3

To detect intersections in $\mathbb{R}^3$, we lift a circle to the corresponding circle in S^3, and look for intersections with (4). Explicitly, a circle in S^3 is the intersection of some plane π with S^3, which we first find. Let the needle arc in $\mathbb{R}^3$ be given in needle-model coordinates by $(r\cos s, r\sin s, 0)$ from $s = 0$ to $s = S$. Name the start point $\mathbf{g}_1 = (r,0,0)$, the end point $\mathbf{g}_2 = (r\cos S, r\sin S)$ and the mid-point $\mathbf{g}_3 = \left(r\cos\frac{S}{2}, r\sin\frac{S}{2} \right)$. Using homogeneous coordinates, the needle arc is mapped into $\mathbb{R}^3$ by a 4×4 position matrix

$$\mathbf{L} = \begin{bmatrix} l_{11} & l_{12} & l_{13} & l_{14} \\ l_{21} & l_{22} & l_{23} & l_{24} \\ l_{31} & l_{32} & l_{33} & l_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

which takes the three points to

$$\mathbf{f}_1 = \mathbf{L}\mathbf{g}_1 = \begin{bmatrix} l_{14}+rl_{11} & l_{24}+rl_{21} & l_{34}+rl_{31} & 1 \end{bmatrix}^T$$

and similarly for $\mathbf{f}_2 = \mathbf{L}\mathbf{g}_2$ and $\mathbf{f}_3 = \mathbf{L}\mathbf{g}_3$. Write their coordinates as (l_1,m_1,p_1,q_1), (l_2,m_2,p_2,q_2) and (l_3,m_3,p_3,q_3) respectively by applying P to $\mathbf{f}_1$, $\mathbf{f}_2$ and $\mathbf{f}_3$. These define the hyperplane

$$\begin{bmatrix} h_1 & h_2 & h_3 & h_4 \end{bmatrix} \begin{bmatrix} l \\ m \\ p \\ q \end{bmatrix} = 0, \quad \text{where}$$

$$h_1 = \begin{vmatrix} m_1 & m_2 & m_3 \\ p_1 & p_2 & p_3 \\ q_1 & q_2 & q_3 \end{vmatrix}, \quad h_2 = -\begin{vmatrix} l_1 & l_2 & l_3 \\ p_1 & p_2 & p_3 \\ q_1 & q_2 & q_3 \end{vmatrix},$$

$$h_3 = \begin{vmatrix} l_1 & l_2 & l_3 \\ m_1 & m_2 & m_3 \\ q_1 & q_2 & q_3 \end{vmatrix}, \quad h_4 = -\begin{vmatrix} l_1 & l_2 & l_3 \\ m_1 & m_2 & m_3 \\ p_1 & p_2 & p_3 \end{vmatrix}.$$

i.e., (h_1,h_2,h_3,h_4) is the triple cross-product of $(l_1,m_1,p_1,q_1) \times (l_2,m_2,p_2,q_2) \times (l_3,m_3,p_3,q_3)$. A vector through the lifted circle's centre is given by the triple cross-product

$$\mathbf{k} = (h_1,h_2,h_3,h_4) \times (\mathbf{f_2}-\mathbf{f_1}) \times (\mathbf{f_3}-\mathbf{f_1}). \tag{7}$$

We adjust the length of $\mathbf{k}$ to get the centre in $\mathbb{R}^4$ of the lifted circle $\widehat{C}$

$$\mathbf{c} = (c_1,c_2,c_3,c_4) = \frac{\mathbf{k} \cdot (l_1,m_1,p_1,q_1)}{\mathbf{k} \cdot \mathbf{k}} \mathbf{k} \tag{8}$$
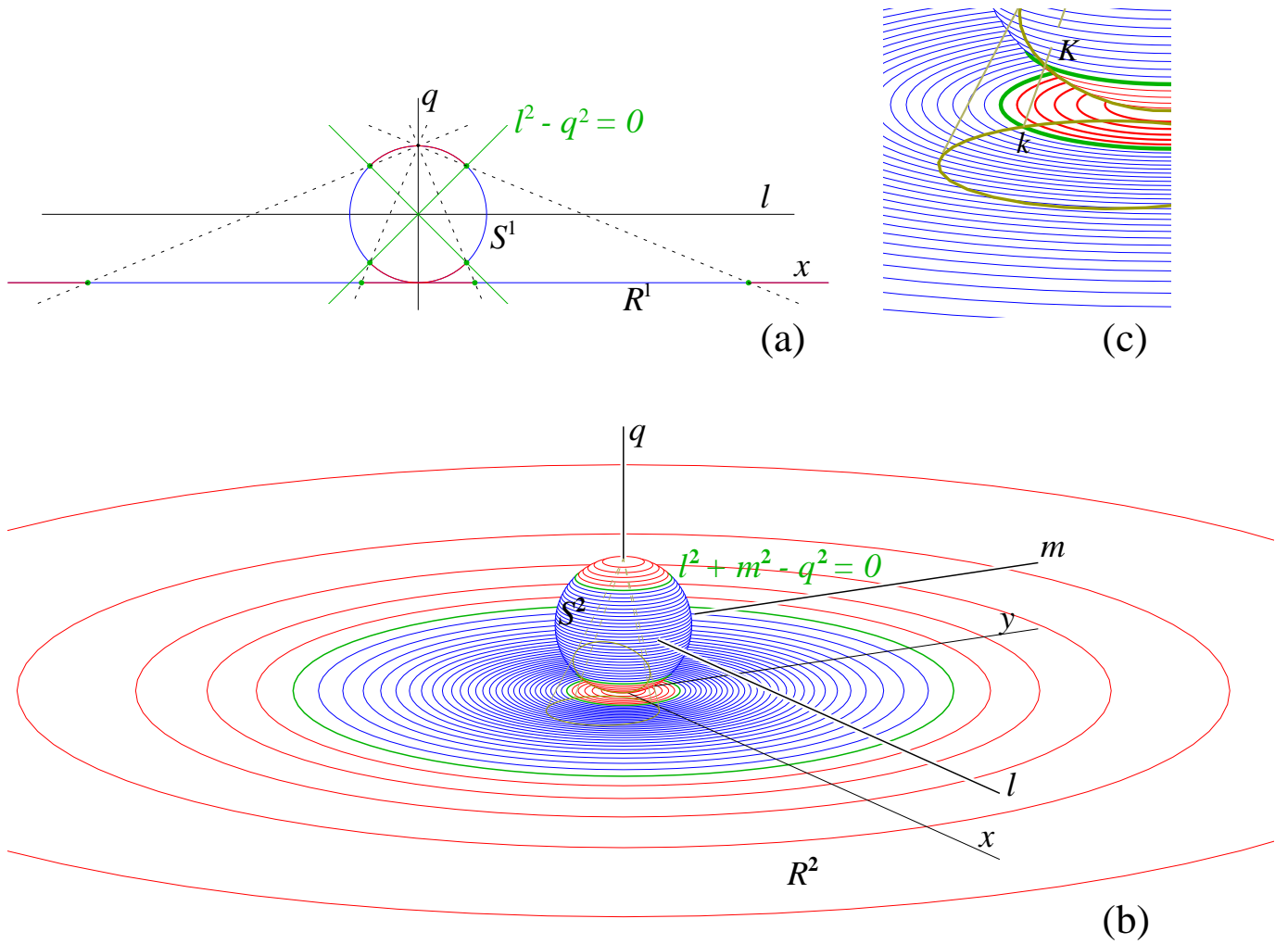
Fig. 6. To detect the intersection of a torus with the needle (circle arc), we lift to S^3 and consider the intersection between the (quadratic) cone corresponding to the torus and the plane defined by the circle on S^3 (§III-B). As it is difficult to represent this in 4 dimensions, this figure illustrates this algorithm in one (a) and two (b,c) dimensions, with the stereographic projection is onto S^2. The detail in (c) shows the correspondence between the intersection k of an arc (needle) with the projected quadric (torus), and the intersection K of its lift with the quadric set in S^3.

The squared radius of $\widehat{C}$ is

$$R^2 = ((l_1, m_1, p_1, q_1) - \mathbf{c}) \cdot ((l_1, m_1, p_1, q_1) - \mathbf{c}). \quad (9)$$

In the plane of $\widehat{C}$ we choose a first unit basis vector

$$\mathbf{b}_x = \frac{1}{R}((l_1, m_1, p_1, q_1) - \mathbf{c}), \quad (10)$$

with coordinates $(b_{xl}, b_{xm}, b_{xp}, b_{xq})$. Orthogonal to this and in the plane π of $\widehat{C}$ is the triple cross-product

$$\mathbf{b}_y^\star = -\mathbf{b}_x \times \mathbf{c} \times (h_1, h_2, h_3, h_4), \quad (11)$$

giving us our second basis vector

$$(b_{yl}, b_{ym}, b_{yp}, b_{yq}) = \mathbf{b}_y \equiv \frac{1}{\sqrt{\mathbf{b}_y^\star \cdot \mathbf{b}_y^\star}} \mathbf{b}_y^\star. \quad (12)$$

A general point (x, y) in π is then

$$\mathbf{c} + x\mathbf{b}_x + y\mathbf{b}_y = (c_1 + xb_{lx} + yb_{ly}, c_2 + xb_{mx} + yb_{my},$$
$$c_3 + xb_{px} + yb_{py}, c_4 + xb_{qx} + yb_{qy}) \quad (13)$$

in $\mathbb{R}^4$ coordinates. If we parametrize the lifted circle by $(R\cos\theta, R\sin\theta)$ the lifted arc still starts (by construction) at $\theta = 0$, but the end angle θ_ω needs not be equal to S. To find it we need to use

$$\theta_\omega = \arctan 2\left(\frac{(\mathbf{f}_2 - \mathbf{c}) \cdot \mathbf{b}_y}{R}, \frac{(\mathbf{f}_2 - \mathbf{c}) \cdot \mathbf{b}_x}{R}\right) \quad (14)$$

In the plane of $\widehat{C}$, the torus (4) becomes

$$0 = ax^2 + bxy + cy^2 + dx + ey + f \quad (15)$$
with

$$a = \alpha^2\left(b_{lx}^2 + b_{mx}^2\right) - b_{qx}^2 - b_{px}^2$$
$$b = \alpha^2\left(2b_{lx}b_{ly} + 2b_{mx}b_{my}\right) - 2b_{qx}b_{qy} - 2b_{px}b_{py}$$
$$c = \alpha^2\left(b_{ly}^2 + b_{my}^2\right) - b_{qy}^2 - b_{py}^2$$
$$d = \alpha^2\left(2c_1 b_{lx} + 2c_2 b_{mx}\right) - 2c_4 b_{qx} - 2c_3 b_{px}$$
$$e = \alpha^2\left(2c_1 b_{ly} + 2c_2 b_{my}\right) - 2c_4 b_{qy} - 2c_3 b_{py}$$
$$f = \alpha^2\left(c_1^2 + c_2^2\right) - c_4^2 - c_3^2$$

D. Computing a distance between the needle and the curved tube

In the lifted circle plane coordinates above, common tangents for contours of

$$T(x,y) = ax^2 + bxy + cy^2 + dx + ey + f \qquad (16)$$
$$C(x,y) = x^2 + y^2$$

occur at the vanishing points

$$\frac{1}{2}\left[\begin{array}{c} \frac{\partial}{\partial x}T(x,y) \\ \frac{\partial}{\partial y}T(x,y) \end{array}\right] \times \left[\begin{array}{c} \frac{\partial}{\partial x}C(x,y) \\ \frac{\partial}{\partial y}C(x,y) \end{array}\right] = 0$$
$$\frac{1}{2}\left[\begin{array}{c} 2ax+by+d \\ bx+2cy+e \end{array}\right] \times \left[\begin{array}{c} 2x \\ 2y \end{array}\right] = 0$$
$$b(x^2 - y^2) + 2(c-a)xy + ex - dy = 0 \qquad (17)$$

Look first at the homogeneous zeroes of

$$b(x^2 - y^2) + (c-a)2xy = 0 \qquad (18)$$

Revolve this [xxx: R2: "revolve this" is improper English] to new (u,v) coordinates, abbreviating one coefficient to $c - a = g$:

$$0 = b(x^2 - y^2) + g(2xy),$$
$$x = u\cos\psi - v\sin\psi, \qquad (19)$$
$$y = u\sin\psi + v\cos\psi$$

yields

$$(u^2-v^2)(b\cos 2\psi + g\sin 2\psi) + 2vu(g\cos 2\psi - b\sin 2\psi) = 0 \qquad (20)$$

If we set

$$0 \equiv b\cos 2\psi + g\sin 2\psi,$$
$$\psi \equiv \frac{-1}{2}\arctan\frac{b}{d}, \qquad (21)$$

(20) becomes

$$2(d\cos 2\psi - b\sin 2\psi)uv = 0 \qquad (22)$$

and the inhomogeneous quadratic for the torus

$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

becomes

$$\beta uv + \gamma u + \delta v = 0 \qquad (23)$$

where

$$\beta \equiv 2(g\cos 2\psi - b\sin 2\psi)$$
$$\gamma \equiv d\cos\psi + e\sin\psi$$
$$\delta \equiv e\cos\psi - d\sin\psi$$

This gives (unless $\beta = \delta = 0$, which reduces (23) to $u = 0$ and we're done)

$$v(\beta u + \delta) + \gamma u = 0 \qquad (24)$$
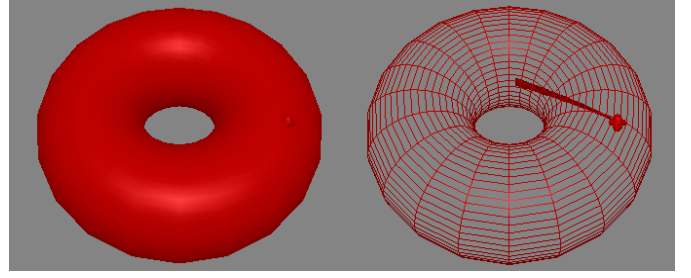$$\frac{-\gamma u}{\beta u + \delta} = v \qquad (25)$$

Fig. 7. The analytical algorithm enables fast computation of the separation of needle and torus.

while $x^2 + y^2 = R^2$ rotates to $u^2 + v^2 = R^2$, so

$$R^2 = u^2 + \left(\frac{-\gamma u}{\beta u + \delta}\right)^2$$
$$0 = u^2(\beta^2 u^2 + 2\beta u\delta + \delta^2 + \gamma^2) - R^2(\beta u + \delta)^2 \qquad (26)$$
$$0 = \beta^2 u^4 + 2\beta\delta u^3 + (\gamma^2 + \delta^2 - \beta^2 R^2)u^2 - 2\beta\delta R^2 u - \delta^2 R^2$$

Get the real roots between $\pm R$ of (26) numerically or with a standard routine [8], plug in to (25), put u and v into (19) and we have the points at which (16) has its extrema[1] on the radius-R circle. We find the θ values of these points in the standard parametrisation $(R\cos\theta, R\sin\theta)$ of the circle.

From this we can determine whether an arc from θ_1 to θ_2 of the radius-R circle meets the set

$$ax^2 + bxy + cy^2 + dx + ey + f = 0. \qquad (27)$$

Writing

$$\phi(\theta) = aR^2(c\theta)^2 + bR^2(c\theta)(s\theta) + R^2 c(s\theta)^2 + Rd(c\theta) + Re(s\theta) + f \qquad (28)$$

with $c\theta \equiv \cos\theta$, $s\theta \equiv \sin\theta$, we find $\phi(0)$ and $\phi(\theta_\omega)$. If there is no maximum or minimum between θ_α and θ_ω, we have an intersection with (27) if and only if 0 is between $\phi(0)$ and $\phi(\theta_\omega)$. If the arc contains n extrema $\theta_1,\ldots,\theta_n$ (for $1 < n \le 4$), there is an intersection if 0 is between $\phi(0)$ and $\phi(\theta_1)$, or between $\phi(\theta_1)$ and $\phi(\theta_2)$ or ... or between $\phi(\theta_n)$ and $\phi(\theta_\omega)$.

Also, the θ of the smallest absolute value of $\phi(\theta)$ between $\phi(0)$ and $\phi(\theta_\omega)$ can be used to find the point of the needle arc which is nearest (in the sense of lowest G) to the surface of the curved tube in $\mathbb{R}^3$. Put $(R\cos\theta, R\sin\theta)$ to (13), we get the coordinate of the nearest point in π. Then use (3) to get the coordinate in $\mathbb{R}^3$. Computing the distance between the nearest point and the core curve of torus decides how far the arc and tube are from collision and whether inside $(G < R)$ or outside $(G > R)$. The distance can be used for warning signals such as force, sounds or color gradients, useful in training.

The discussion above applies to the whole torus, to simplify the first reading and checking. For a segment of torus, contained between the planes $\{(x,y,z), x = 0\}$ and $\{(x,y,z), (\cos\Omega)x + (\sin\Omega)y = 0\}$, it is straightforward to find the part of the needle arc that also lies between the two planes,

---

[1]Which are minima and which are maxima appears easily, by inspecting the values. If there are two solutions, the greater and lesser are the unique maximum and minimum. If there are four, both maxima are above both minima. However, we do not need this information.
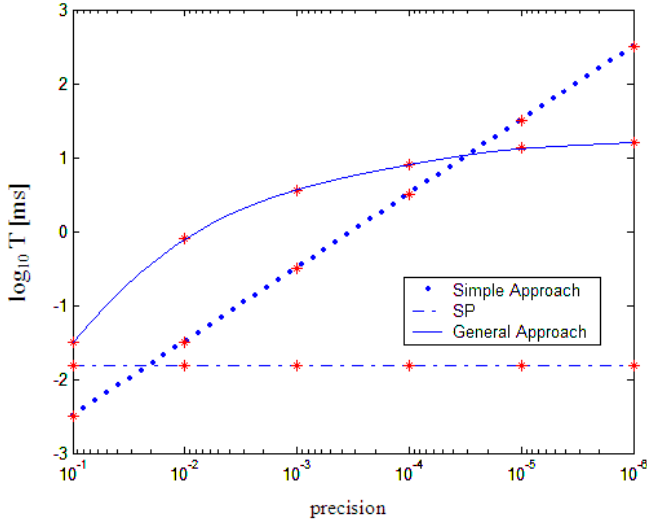
Fig. 8. Computation time for torus-needle separation for the analytic method versus two numerical ones, comparing to the three approaches of (Fig. 5).

and test that rather than the whole arc for collision with the torus.

IV. PERFORMANCE AND COMPARISON

We implemented the *SP* algorithm of section III (on a Dell PC with 2.4GHz Pentium CPU, 256 MB) to detect collision between a circular needle and a toroidal blood vessel (Fig. 7). We measured the computation time as a function of the accuracy, defined as the end criterion of the Newton algorithm used to solve the fourth order equation (26). We see in Fig. 8 that the computation time is as low as $0.015ms$ and does not increase with accuracy, an important property to simulate the microworld.

The *SP* algorithm was first compared with a general detection algorithm, the RAPID interference detection system [22]. For a fair comparison we excluded the set-up time of this algorithm, as well as the visualization effort which common to all the algorithms. The precision of RAPID depends on the diameter of the largest triangle. This is a slightly different measure of precision than for *SP*, however the computation time is order of magnitudes larger than with *SP*, so this difference is inessential. Similarly, the algorithm of discretizing the needle and computing the distance to the torus for all these discrete points (Fig. 5b) requires computation time growing exponentially with the precision. In that case the precision corresponds to the discretization step.

In conclusion, the *SP* algorithm is orders of magnitude faster than numerical approaches at accuracies below $10^{-xxx}m$, and can be used to implement all audio, visual and haptic cues of section II-C.

V. DISCUSSION

We are developing a Virtual Reality micromanipulation learning system to train microsuturing, implemented on the dedicated workstation [3], [13] with collocated visual and haptic workspaces [25].

Conventional approaches judge a virtual environment by its resemblance to the real environment, and often the 'sense of immersion' or 'presence'. In contrast, we are only concerned with efficiency in learning, and immerse the hands rather than the head[xxx what is better: head or brain?]. We train simple dexterity primitives corresponding to selected features of the complex task, and also investigate suitable multi-sensory learning cues systematically. To this end we use simple environments with only selected features of the complex tasks, and develop fast algorithms to investigate the learning of these dexterity primitives with various multi-sensory cues.

For the needle maneuvering skill, we introduced a method using stereographic projection to compute the distance between the curved needle and a curved vessel. This analytical algorithm was shown to be faster by orders of magnitude than numerical ones. Furthermore the computation time barely increases with increasing precision, a critical condition to simulate the microworld. We used geometric properties specific to the shapes involved, such as the circularity of the arc allowing it to lift to a planar circle in S^3, as an ellipse for instance does not. This limits the generality of the specific algebra used, but still shows that geometrically special coding can be as powerful as rewriting an algorithm in system-specific assembler for speedup. [xxx R2: The last two sentences have awkward grammar]

APPENDIX: RELATIONS BETWEEN R, r AND α

To connect the torus form (1) with the projected α version, we substitute

$$l = \frac{2x}{x^2+y^2+z^2+1} \quad m = \frac{2y}{x^2+y^2+z^2+1}$$
$$p = \frac{2z}{x^2+y^2+z^2+1} \quad q = \frac{x^2+y^2+z^2-1}{x^2+y^2+z^2+1}$$

from (3) into (4). We have $\alpha^2\left(l^2+m^2\right) - \left(p^2+q^2\right) = 0$ exactly at the zeroes of $\varepsilon\left(x^2+y^2+z^2+1\right)^2\left(\alpha^2\left(l^2+m^2\right)-\left(p^2+q^2\right)\right)$ where ε is an arbitrary constant. We want to choose R and r such that

$$\left(\rho^2-D^2\right)^2-4\left(x^2+y^2\right)R^2=C\left(\left(\rho^2+1\right)^2\left(\alpha^2\left(l^2+m^2\right)-\left(p^2+q^2\right)\right)\right)$$

where $\rho^2 = x^2+y^2+z^2$, $D^2 = R^2-r^2$. But then

$$\rho^4-2\rho^2D^2+D^4-4\left(x^2+y^2\right)R^2-C\left(\alpha^2\left(4x^2+4y^2\right)-4z^2-\left(\rho^2-1\right)\right)$$

vanishes at infinitely many points, which is only true if the coefficients of x^n, y^n, z^n vanish identically. Comparing the coefficients of ρ^4, we want $\varepsilon = -1$, *i.e.*,

$$\left(2D^2-4R^2+2+4\alpha^2\right)x^2+\left(2D^2-4R^2+2+4\alpha^2\right)y^2+\left(2D^2-2\right)z^2+D^4-1=0$$

The vanishing of coefficients reduces to

$$D^4 - 1 = 0 \tag{29}$$
$$2D^2 - 2 = 0$$
$$2D^2 - 4R^2 + 2 + 4\alpha^2 = 0$$

We want a torus with $R > r$, so we choose

$$R^2 - r^2 = 1. \tag{30}$$

It then follows that

$$4R^2 - 4\alpha^2 = 4 \tag{31}$$
$$R^2 - \alpha^2 = 1$$
$$r = \alpha$$

For small values of α we get a thin torus containing the unit circle $x^2 + y^2 = 1$ in the $z = 0$ plane but centred on $x^2 + y^2 = 1 + \alpha^2$. For large values we get a fat torus, with a z-axis hole of throat radius

$$\sqrt{1 + \alpha^2} - \alpha = \alpha \left(\sqrt{1 + \frac{1}{\alpha^2}} - 1 \right) \tag{32}$$
$$= \alpha \left(\left(1 + \frac{1}{2\alpha^2} + \dots \right) - 1 \right)$$
$$\approx \frac{1}{2\alpha}$$

and outside radius approximately α.

Note that (30) forces $r = \sqrt{R^2 - 1}$, regardless of α. To get a general torus, with arbitrary $\widetilde{R}$ and $\widetilde{r}$, define their ratio $\beta = \widetilde{r}/\widetilde{R}$. We produce a torus with this ratio by setting

$$\beta^2 = \frac{\alpha^2}{1 + \alpha^2} \tag{33}$$
$$\alpha^2 = \frac{\beta^2}{1 - \beta^2},$$

which gives $R^2 = 1/(1 - \beta^2)$. If this is not the value $\widetilde{R}$ needed, scale (x, y, z) by the factor $\widetilde{R}/\sqrt{1 - \beta^2} = \sqrt{\widetilde{R}^2 - \widetilde{r}^2}$. For instance, if $\widetilde{R} = 5$ and $\widetilde{r} = 3$, giving $\beta = 3/5$, we want to draw the $\alpha^2 = 9/16$ torus (which unscaled would have $R = \sqrt{1 + 9/16} = 5/4$ and $r = \sqrt{9/16} = 3/4$) enlarged by the factor $\sqrt{5^2 - 3^2} = 4$.

REFERENCES

[1] A Dhanik, T Poston, KM Lim, CL Teo, E Burdet, Virtual nylon thread manipulation for a haptic microsurgery training environment. Submitted. 2003.

[2] C. Shene and J. Johnstone. On the planar intersection of natural quadrics. Proceedings of ACM Solid Modeling, 234–244, 1991.

[3] CL Teo, E Burdet and P Lim (2002), A Robotic Teacher of Chinese Ideograms, Proc. IEEE Virtual Reality (IEEEVR).

[4] E Burdet, R Osu, DW Franklin, T Milner and M Kawato (2001), The central nervous system stabilizes unstable dynamics by learning optimal impedance, Nature 414: 446–449.

[5] ES Boy, E Burdet, CL Teo and JE Colgate, Experimental Evaluation of the Learning Cobot. Proc. Eurohaptics, 2003.

[6] HD Zimmer et al. (2000), Memory for Action: A Distinct Form of Episodic Memory? Counterpoints, Oxford University Press.

[7] H Zhang, E Burdet et al. (2002), Robotic Micro-assembly of a scaffold/cell constructs with a shape memory alloy gripper, Proc IEEE International Conference on Robotics and Automation.

[8] http://www.uni-koeln.de/math-nat-fak/phchem/deiters/quartic/quartic.html

[9] J Brown, J-C Latombe and K Montgomery. Real-time Knot Tying Simulation. Dimacs Workshop on Medical Applications in Computational Geometry, Rutgers, New Jersey, April, 2003.

[10] J Brown, K Montgomery, J-C Latombe, and M Stephanides. A Microsurgery Simulation System. Medical Image Computing and Computer-Assisted Intervention (MICCAI 2001) in Utrecht, The Netherlands, October 2001.

[11] LV Ahlfors, Complex Analysis - An Introduction to the Theory of Analytic Functions of One Complex Variable. McGraw Hill 1979.

[12] JR Miller, Geometric approach to nonplanar quadric surface intersection curves. ACM Transactions on Graphics, 6(4): 274-307, 1987.

[13] L Bingfeng (2002) A manipulation simulator with 3-D visual and 6DOF haptic feedback, MEng thesis, National University of Singapore.

[14] M Haruno, DM Wolpert and M Kawato (2001), MOSAIC Model for Sensorimotor Learning and Control, Neural Computation 13: 2201-2220.

[15] M Kawato (1999), Internal Models for Motor Control and Trajectory Planning, Current Opinion in Neurobiology 9: 718–727.

[16] N Bhushan and R Shadmehr (1999), Computational nature of human adaptive control during learning of reaching movements in force fields, Biological Cybernetics 81(1): 39–60.

[17] L Piegl (1989), Geometric method of intersecting natural quadrics represented in trimmed surface form, Comput. Aided Des. 21(4):201–212.

[18] RD Shilling, B G Shinn-Cunningham (2001), Virtual auditory displays, in Handbook of Virtual Environment Technology, K. Stanney ed.

[19] RN Lemon, RS Johansson and G Westling (1995) Corticospinal control during reach, grasp, and precision lift in man, the Journal of Neuroscience 15: 6145–6156

[20] RS Johansson et al. (2001) Eye-hand coordination in object manipulation, the Journal of Neuroscience 21: 6917–6932

[21] RT Farouki, CA Neff and M O'Connor. Automatic parsing of degenerate quadric-surface intersections. ACM Transactions on Graphics 8: 174-203, 1989.

[22] S Gottschalk, M Lin and D Manocha. OBB-tree: A hierarchical structure for rapid interference detection. In Proc. ACM Siggraph'96 171-180, 1996.

[23] S Smith, RM Pickett and MG Williams (1994), Environments for Exploring Auditory Representations of Multidimensional Data. Auditory Display: Sonification, Audification, and Auditory Interfaces, G Kramer ed., Santa Fe Institute Studies in the Sciences of Complexity, Proc. Vol. XVIII. Reading, MA: Addison-Wesley.

[24] T Flash and E Henis (1991), Arm Trajectory modifications during reaching toward visual targets, Journal of cognitive Neuroscience 3(3): 220–230.

[25] T Poston (1997), The Virtual Workbench: A Path to Use of VR, in The Industrial Electronics Handbook, JD Irwing ed., IEEE Press, 1390–1393.