Paper Number 2006-01-2320

Session DHM5

The Virtual Movable Human Upper Body for Palpatory Diagnostic Training

Meng-Yun Chen, Robert L. Williams II, Robert R. Conatser Jr. and **John N. Howell**
Interdisciplinary Institute for Neuromusculoskeletal Research, Ohio University

ABSTRACT

This paper presents our virtual model for the dynamic human upper body, including the spine, shoulders, and arms skeletal structure. We enable realistic human motion with anatomically-accurate joint motion limits and a 71 degrees-of-freedom branching serial chain model. Our interest is to provide realistic motions when the student doctor moves the virtual patient for palpatory diagnosis training. To date, this work include only skeletal motion with skin stretched on the model, but in the future we will augment our static Virtual Haptic Back model with this motion. In this way, student doctors can feel changes in human tissue due to motions, a common diagnostic technique.

KEYWORDS

Virtual Haptic Back, virtual movable human upper body, palpatory diagnosis, haptics, skeletal motion, biomechanics

INTRODUCTION

Most virtual human research focuses on workspace and obstacle avoidance analysis [1, 2, 3]. Human limbs are considered as the end-effectors. The majority of virtual human research studies the results of the end-effectors movement. The Virtual Haptic Back (VHB) focuses on individual vertebra movement rather than the work space of the limbs. However, the individual bone movement is as a result of end-effectors movement. The intermediate joint angles of the virtual haptic back model were calculated in real time and using robotics methods rather than the interpolation method of [4].

The Virtual Haptic Back (VHB) is under development at Ohio University as a teaching and learning tool for medical palpatory diagnosis (finding disease and somatic dysfunction via touch). Two PHANToM 3.0 haptic interfaces permit palpation of a life-sized virtual human back. A graphics image of the back is displayed and local motion of back components (skin or underlying vertebrae) by exertion of palpatory force by the user is reflected graphically. Mechanical properties of the back, e.g., spring constants of the surface, are chosen in part based on measurements and in part based on feedback from physicians experienced in palpatory diagnosis.

To date our VHB model is static (other than the local motion described above); that is, there is no gross motion of the simulated human under diagnosis. A common technique used by palpating physicians is to impart gross motion of the human upper body to create spinal movement in order to find more easily and accurately abnormal vertebrae and surrounding tissues. This paper summarizes our efforts to create a user-movable virtual human upper body. Our model includes 71 degrees-of-freedom, with the human spine, shoulders, and arms, having realistic graphics and anatomically-accurate joint motion limits. We use robotics methods and equations to produce a surprisingly-realistic virtual human motion. We have also covered this model with dynamic skin, allowing us in the near future to use haptics technology to enable medical students to feel realistic differences in touch while using the gross motion technique. This paper consist of five sections: modeling the human spinal column and upper body, dynamics, deformable skin, haptics and software development.

MODELING THE HUMAN SPINAL COLUMN AND UPPER BODY

The human skeleton can be considered as series of rotational links connected together. The anatomy of a human spine is shown in Figure 1. This research focuses on the thoracic vertebrae, lumbar vertebrae, shoulders and two arms. A 3D human shoulder model is shown in Figure 2. To describe the translational and rotational relationships between adjacent links of the open kinematic chains, Denavit and Hartenberg (DH) notation [5] has been used because of its strength in handling large numbers of degrees of freedom and because of its ability to systematically enable kinematic and dynamic analyses.

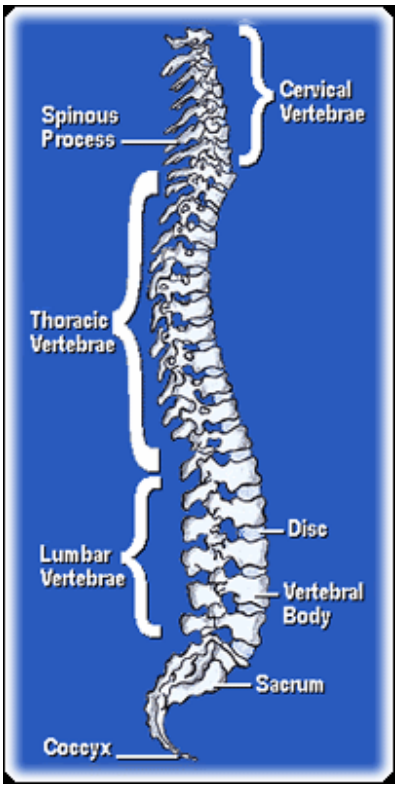


Figure 1. Anatomy of the human spine

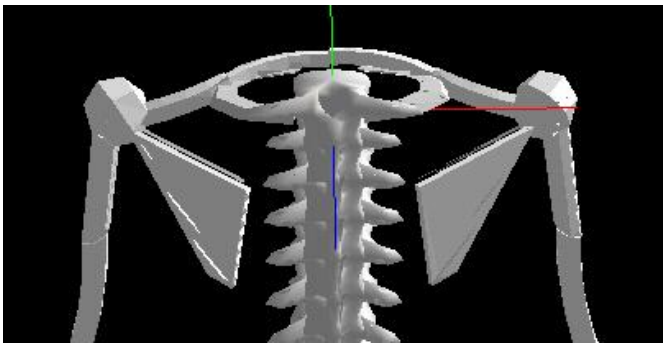


Figure 2. 3D human shoulder model

Consider Figure 3 where two consecutive joints are shown.

The four DH parameters depicted in Figure 3 are:

a_i distance from Z_i to Z_{i+1} measured along X_i
α_i angle between Z_i and Z_{i+1} measured about X_i
d_i distance from X_{i-1} to X_i measured along Z_i
θ_i angle between X_{i-1} to X_i measured about Z_i

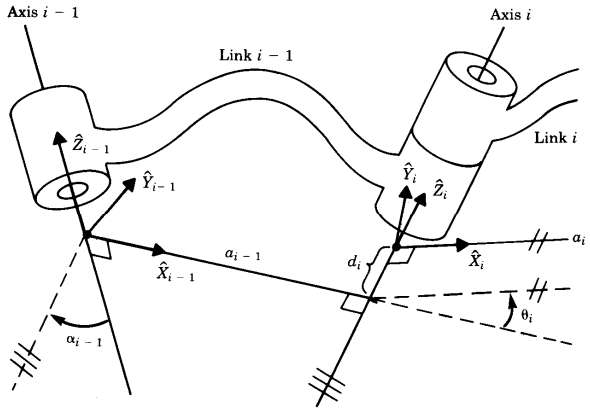


Figure 3. Joint reference frames for D-H representation (taken from [1])

The position vector of a point of interest on the end-effector (any point or points of interest, where the student doctor can hold and move, such as the elbow) of a human articulated model can be written in terms of joint coordinates as

$$X=X(\theta)$$

where $\theta=[\theta_1...\theta_n]^T \in R^n$ is the vector of n generalized joint coordinates defining the motion of a link with respect to its previous neighbor in the serial chain. The global position vector $X(\theta)$ can be obtained from the multiplication of the 4x4 homogeneous transformation matrices ${}^{i-1}_{i}T$, defined by the D-H representation method [1] as follows:

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

The 4x4 transformation matrix ${}^{0}_{i}T$ used to represent i^{th} joint coordinate system with respect to the global base coordinate system {0} is:

$${}^{0}_{i}T(\theta)={}^{0}_{1}T(\theta_1){}^{1}_{2}T(\theta_2)...{}^{i-1}_{i}T(\theta_i)$$

We use the augmented 4x1 vectors ${}^{0}_{i}r$ and ${}^{i}_{i}r$ to express the Cartesian coordinates of a point fixed in the i^{th} local frame in terms of the global coordinate system, respectively:

$${}^{0}_{i}r=\begin{bmatrix} X(\theta) \\ 1 \end{bmatrix} \qquad {}^{i}_{i}r=\begin{bmatrix} {}^{i}X \\ 1 \end{bmatrix}$$

where ${}^{i}X$ is the fixed point of interest on link i, expressed with respect to the i^{th} coordinate system. Using these relationships, ${}^{0}_{i}r$ can be written as:

$$ {}^{0}_{i}r = {}^{0}_{i}T(\theta)\,{}^{i}_{i}r $$

Given the set of joint angle θ, we can calculate the end-effector coordinates *X*. This is also called forward pose (position and orientation) kinematics, which is a straight-forward computation:

$$ X = f(\theta) $$

The Virtual Haptics Human Upper Body is focused on moving the end-effector in order to find all the joint angles. This procedure requires the solution of the inverse pose kinematics problem:

$$ \theta = f^{-1}(X) $$

Our Virtual Haptics Human Upper Body model consists of 71 degrees-of-freedom (DOF). The end-effector has six DOF (*x*,*y*,*z*,*Rx*,*Ry*,*Rz*). The system joint DOF is much greater than the end-effector Cartesian DOF. Our model is a hyper-redundant system. A redundant system has an infinite number of solutions for joint angles in the inverse pose kinematics problem.

In our case, we used Jacobian-based Inverse Kinematics to obtain the joint angles at every simulation step. This is an effective method to handle our hyper-redundancy with 71-6 = 65 redundant joint freedoms. The results yield surprisingly human-like motions. The following bullets describe a Jacobian matrix:

- A Jacobian is a vector derivative with respect to another vector, so it is a multi-dimensional form of the derivative.
- If we have a vector-valued function of a vector of variables $f(\theta)$, the Jacobian is a matrix of partial derivatives, with one partial derivative for each combination of components of the vectors (the rows represent the functions *f* and the columns indicate the variables θ).
- The Jacobian matrix thus contains all of the information necessary to relate a change in any component of *f* to a change in any component of θ.
- The Jacobian is usually written as $J(f,\theta)$, and it is generally represented as the matrix $[\partial f/\partial\theta]$:

$$ J(f,\theta) = \frac{\partial f}{\partial \theta} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} & \cdots & \frac{\partial f_1}{\partial \theta_n} \\ \frac{\partial f_2}{\partial \theta_1} & \frac{\partial f_2}{\partial \theta_2} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_m}{\partial \theta_1} & \cdots & \cdots & \frac{\partial f_m}{\partial \theta_n} \end{bmatrix} $$

is a Jacobian matrix, where in our case *m* = 6 Cartesian DOF and *n* = 71 joint DOF.

The Jacobian-based Inverse Kinematics can be expressed as:

$$ d\theta = J^{-1}dX $$

dX is the change of the end-effector
$d\theta$ is the change of joint angles in the system

However, J^{-1} is not always obtainable. For a hyper-redundant system, J is not a square matrix and cannot be inverted. If we have a non-square matrix arising from a hyper-redundant system, we can use the underconstrained *pseudoinverse* [6, 7, 8]:

$$ J^{+} = (J^{T}J)^{-1}J^{T} $$

This is a method for finding a matrix that effectively inverts a non-square matrix. Our Virtual Haptics Human Upper Body program uses Jacobian Inverse Kinematics to acquire the changes of joint angles in every simulation step according to $d\theta = J^{+}dX$.

DYNAMICS

The dynamic analysis of the Virtual Haptics Human Upper Body is to find the force and torque acting on each joint by the external load acting on the end-effector. We choose iterative Newton-Euler dynamic formulation [1] to obtain the forces and torques. We solve the inverse dynamics problem directly with the standard general method in [5] to solve for the joint forces and torques given the desired human upper body motion. This work is still in progress.

DEFORMABLE SKIN

The Virtual Haptics Human Upper Body will be implemented with deformable skin for a realistic deformation according to the bone movement underneath the skin. This is critical to our future haptics feel, enabling the student to feel realistic tissue texture changes due to motion of the virtual patient.

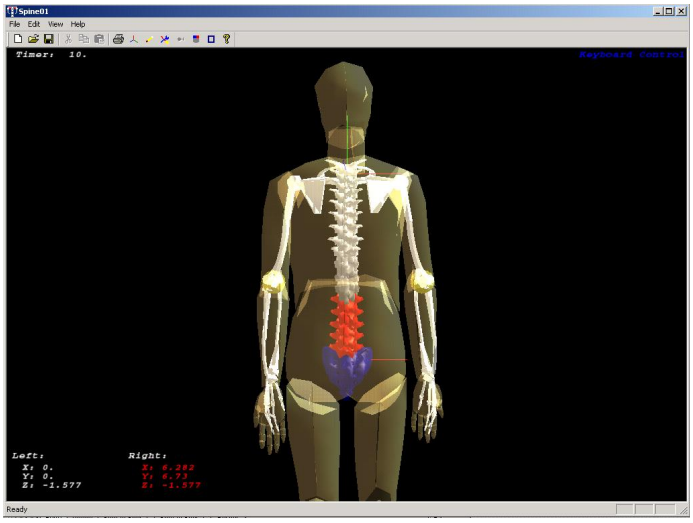


Figure 4. Virtual Haptics Human Upper Body with skin attached

Robots and simple characters made up from a collection of rigid components can be rendered through classical hierarchical rendering approaches [9, 10]. Each component mesh is simply transformed into world space by the appropriate joint world matrix. This results in every vertex in the final rendered character being transformed by exactly one matrix.

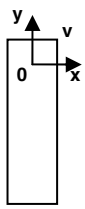
Expressed mathematically, we can say that for every vertex, we compute the world space position **v'** by transforming the local space position **v** by the appropriate joint world matrix **W**:

$$\mathbf{v}' = \mathbf{v} \cdot \mathbf{W}$$

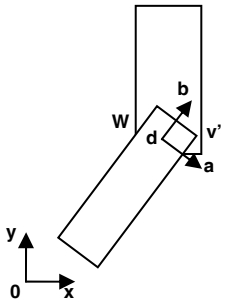
where **W** is a 4x4 matrix and **v** is a 1x4 homogeneous position vector:

$$\mathbf{v} = \begin{bmatrix} v_x & v_y & v_z & 1 \end{bmatrix}$$

Every vertex in each mesh is transformed from the joint local space (where it is defined) into the world space, where it can be used for further graphical processing such as lighting and rendering.



Vertex **v** defined in joint's local space

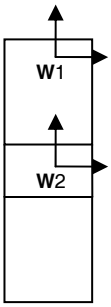


Vertex **v** transformed to world space using matrix **W**

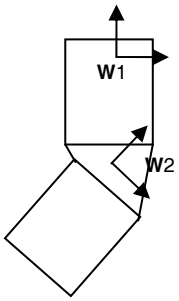
Rendering with rigid components works just fine for robots, mechanical characters, and vehicles, but it is clearly not appropriate for organic characters with continuous skin.

With the *simple skinning* approach [11], the character's skin is modeled as a single continuous mesh. Every vertex in the mesh is attached to exactly one joint in the skeleton, and when the skeleton is posed, the vertices are transformed by their joint's world space matrix. As with the rigid component method, every vertex is transformed by exactly one matrix using an identical equation: $\mathbf{v}' = \mathbf{v} \cdot \mathbf{W}$. This implies that simple skinning should run about the same speed as rendering a character as rigid parts, and in practice, the two techniques often perform similarly with equal-sized meshes.

The simple skinning technique is adequate for low-detail models, but is not sufficient for higher-quality characters. In practice, the simple skinning algorithm can be made to work for characters with perhaps 500 or even as many as 1000 triangles, as long as care is taken in vertex placement and bone attachment. Simple skinning may be sufficient for lower-detail characters, but for higher quality, it is too limited and a better solution is required.

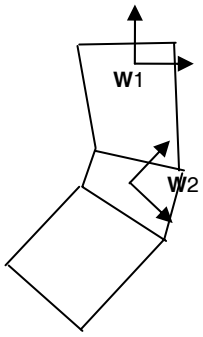


An unbent knee with skin attached to joints 1 and 2



Every vertex is attached to exactly one joint, so as the knee bends, we get some distortion

Smooth skin [12, 13] extends the concepts used in simple skin. With smooth skinning, each vertex in the mesh can be attached to more than one joint, each attachment affecting the vertex with a different strength or *weight*. The final transformed vertex position is a weighted average of the initial position transformed by each of the attached joints. For example, the vertices in a character's knee could be partially weighted to both the hip joint (controlling the upper thigh) and knee joint (controlling the calf). Many vertices will only need to be attach to one or two joints and rarely is it necessary to attach a vertex to more than four.



Using smooth skin, a vertex can be attached to more than one joint with and receive a weighted average of the transformations

Let us say that a particular vertex is attached to *N* different joints. Each attachment is assigned a weight w_i which represents how much influence the joint will have on it. To ensure that no undesired scaling will occur, we enforce the constraint that all of the weights for a vertex must add up to 1:

$$\sum w_i = 1 \qquad = w_0 + w_1 + \ldots + w_{N-1}$$

To compute the world space position **v'** of the vertex, we transform it by each joint that it is attached to, and compute a weighted sum of the results:

$$\mathbf{v}' = \sum w_i \mathbf{v} \cdot \mathbf{B}_{[i]}^{-1} \cdot \mathbf{W}_{[i]}$$

where **v** is the untransformed vertex in *skin local space*, the space in which the skin mesh was originally modeled. The matrix $\mathbf{W}_{[i]}$ is the world matrix of the joint for attachment *I*, resulting from forward pose kinematics computations. We use the indexing notation [*i*] to indicate that we don't want the matrix of the i^{th} joint in the skeleton (which would be written $\mathbf{W}_i$), but instead we want the world matrix of attachment *i*'s joint. For example, if a particular vertex is weighted 60% to joint #37, 30% to joint #6, and 10% to joint #14, then we have:

N=3

$$\mathbf{W}_{[0]} = \mathbf{W}_{37}, \mathbf{W}_{[1]} = \mathbf{W}_{6}, \text{ and } \mathbf{W}_{[2]} = \mathbf{W}_{14}$$

$$w_0 = 0.6 \qquad w_1 = 0.3 \qquad w_2 = 0.1$$

The matrix $\mathbf{B}_{[i]}$ is called the *binding matrix* for joint [*i*]. This matrix is a transformation from joint local space to skin local space, and so the inverse of this matrix, $\mathbf{B}_{[i]}^{-1}$, represents the opposite transformation from skin local space to joint local space. The combined transformation $\mathbf{B}_{[i]}^{-1}\ \mathbf{W}_{[i]}$ therefore first transforms **v** from skin local to joint local, then from joint local to world space. As the number of joints is likely to be small compared to the total number of vertices that need to be skinned, it is more efficient to compute $\mathbf{B}_{[i]}^{-1}\ \mathbf{W}_{[i]}$ for each joint before looping through all of the vertices. We will call this transform $\mathbf{M}_{[i]}$, defined by:

$$\mathbf{M}_{[i]} = \mathbf{B}_{[i]}^{-1} \cdot \mathbf{W}_{[i]}$$

The skinning equation that must be computed for each vertex then simplifies to

$$\mathbf{v}' = \sum w_i \mathbf{v} \cdot \mathbf{M}_{[i]}$$

With the smooth skin algorithm, the attached skin can react realistically and automatically according to the underlying bone movement.

HAPTICS

The Virtual Haptics Human Upper Body implementation will combine two haptics devices. The Immersion Cyber Grasp Haptic Glove (Figure 5) will be used for grasping the virtual human upper body and moving the virtual human upper body (while feeling the shape of the human body where grasped).

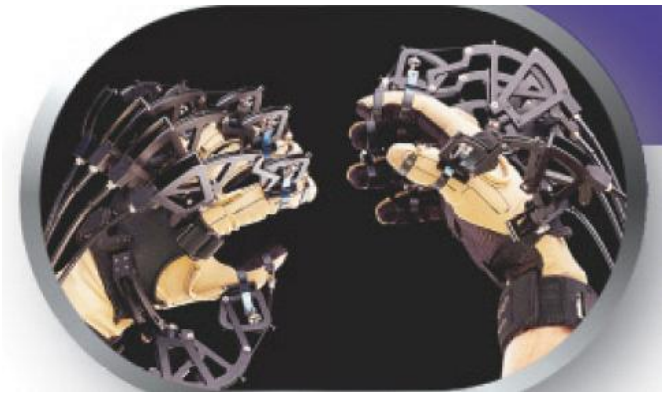


Figure 5. Cyber Grasp haptics glove

The second haptics device is the PHANToM 3.0 (Figure 6) which will be used to feel the spinal processes and soft tissue. The feel will change automatically in a realistic manner (as determined by palpatory expert D.O.s) as the virtual patient moves.

Figure 6. PHANToM 3.0

SOFTWARE DEVELOPMENT

The program for the Virtual Haptics Human Upper Body is written using Visual C++ and the OpenGL graphic library for 3D graphics. Later, the haptic effects will be added by using the Virtual Hand SDK and the PHANToM Ghost SDK. The program will run in a Microsoft Windows environment.

Figure 7 shows a screen capture of the program. Figure 8 is another view of the program. The user can navigate (rotate, tilt, zoom) the graphics in a 3D environment.

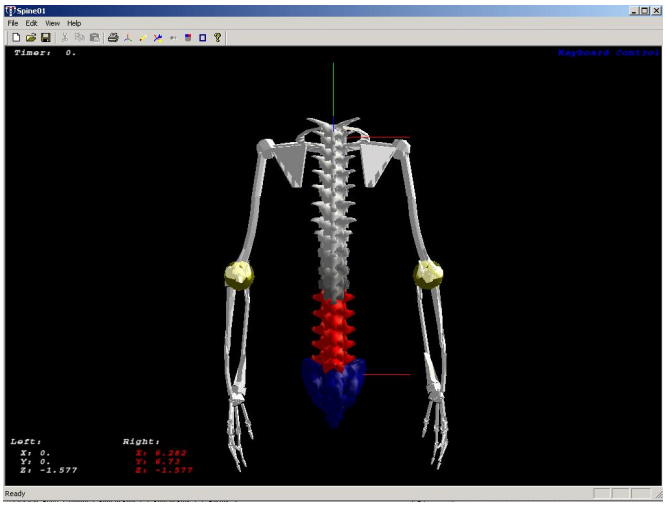


Figure 7. A screen capture of the program

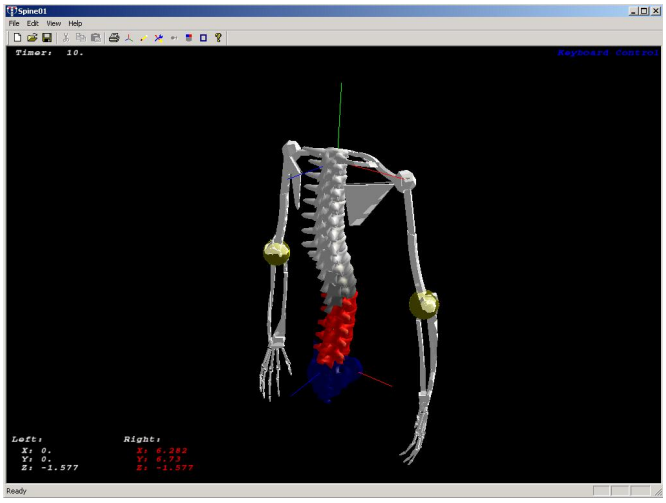


Figure 8. Another view of the program

Figure 9 is the screen capture of the program settings that includes user definable joint limits (set by default to standard anatomical limits), point of interest (one or two, also called end-effector) selection and enable or disable path following. The data of the default joint limits is gathered from Kapandji [14, 15] and White and Panjabi [16]. The user can select point(s) of interests along the spine, shoulder and two arms to move. Then our inverse pose kinematics automatically calculates the motion of the virtual upper body. Figure 10 shows there are two drop-down lists in the program settings by which the user can choose any left and/or right points of interest. Another unique feature of the Virtual Haptic Human Upper Body program is user definable paths for the end-effector (point of interest). Figure 11 shows a user-created path (3D inputs by mouse clicks to draw control points for the path) and the created path is shown in Figure 12. Figure 12 also shows the end-effector (point of interest) following the user-created path, with realistic human motion according to the inverse pose kinematics solution at each motion step. The Haptic (force feedback) effects will be programmed into the Virtual Haptic Human Upper Body by using Virtual Hand SDK and Phantom Ghost SDK in the near future.

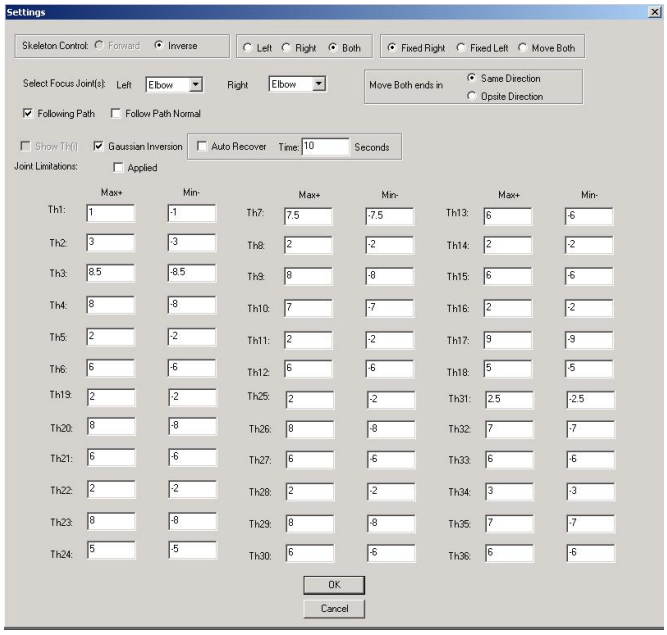


Figure 9. A screen capture of the program settings

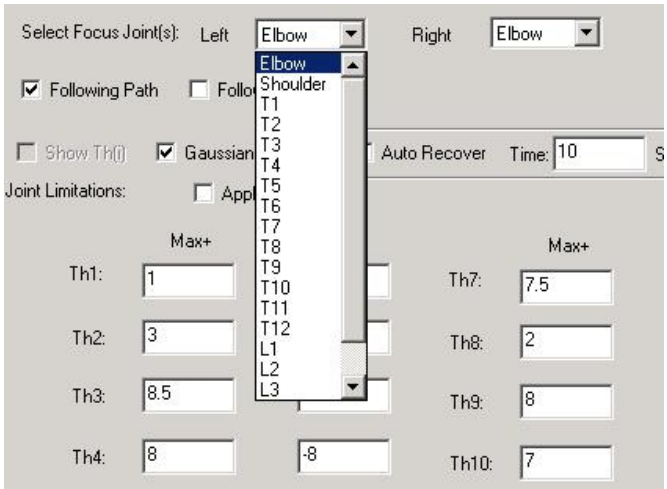


Figure 10. Choose point(s) of interest.

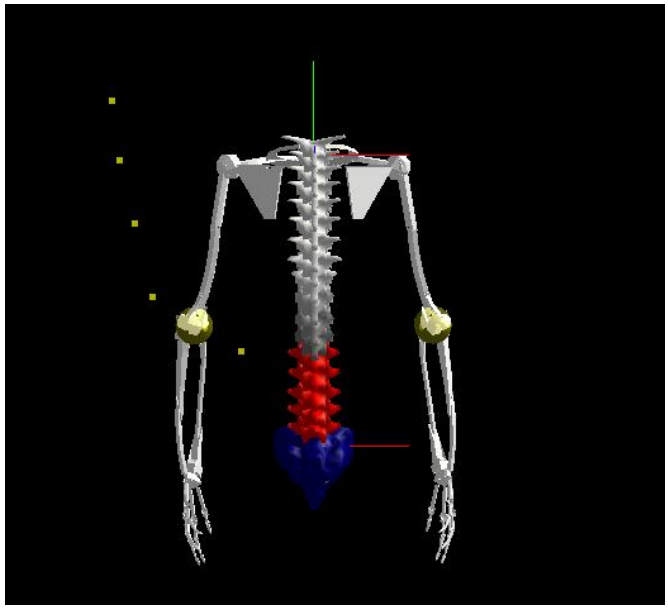


Figure 11. User-created path for the end-effector (elbow)

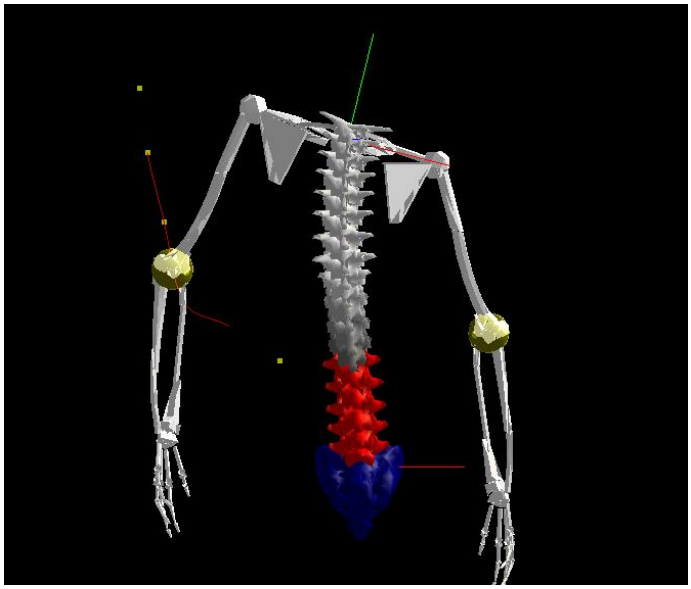


Figure 12. Elbow following the user-created path

VALIDATION AND VERIFICATION

Currently we have no validation and verification data. This section explains our validation and verification plans, since realism is crucial to our work and philosophy. We collaborate with the Ohio University Physical Therapy Department, who maintains a human body motion capture laboratory. We plan to use this lab with an expert palpator moving human subjects through standard palpatory diagnosis motions (such as side-bending and flexion-extension). The motion capture data will be compared with the same expert palpator moving the virtual patient in the virtual world. Errors will be noted and the virtual model will be improved in order to reduce errors.

Now, there are some problems with this validation and verification plan. Currently, it is not nearly as natural for the doctor to move the virtual patient in the same way she moves the real-world patient. Thus, some component of the error will be due to inability to exactly replicate the real-world motions in the virtual world. Also, the motion capture laboratory can only record motion of marked points on the surface of the human subject, whereas we would like to verify human motion down to the vertebral level, which is not possible in the proposed lab work. Nonetheless, we believe that this proposed validation and verification will help improve the realism of our Virtual Haptics Human Upper Body and lead to ideas for future work.

CONCLUSION

The Virtual Haptics Human Upper Body is currently under development. The purpose is to generate automatic and realistic human motion for our Virtual Haptic Back project, in which student doctors are trained in their sense of touch for palpatory diagnoses. This work is a major addition to our work since it represents the first time our virtual human patient can move in a global sense, which changes the palpatory feel of vertebrae and soft tissues. We incorporate forward and inverse pose kinematics solutions and user-selectable end-effector(s) into our program. Another feature is user-definable paths. This feature can be used for validating our model with biomechanical human data in the future. Soft skin, dynamics and haptic feedback will be implemented in the near future.

ACKNOWLEDGMENTS

The authors gratefully acknowledge funding for this work from the Osteopathic Heritage Foundation.

REFERENCES

1. M.P. Reed, K. Satchell, and A. Nichols, 2005, "Application of Digital Human Modeling to the Design of a Postal Delivery Vehicle", Digital Human Modeling for Design and Engineering Symposium.

2. W. Maurel and D. Thalmann, "Human Shoulder Modeling including Scapulo-Thoracic Constraint and Joint Sinus Cones", C&G 2000-Computer and Graphics Vol. 24, No. 2, pp. 203-218.

3. J.K. Hodgins, W.L. Wooten, D.C. Brogan, and J.F. O'Brien, "Animating Human Athletics", Proceedings of SIGGRAPH 95, August, 1995, pp. 71 - 78.

4. K. Abdel-Malek, J. Yang, Z. Mi, V.C. Patel, K. Nebel, 2004, "Human Upper Body Motion Prediction", Applied Simulation and Modeling (ASM), June 28-30.

5. J.J. Craig, 2005, Introduction to Robotics: Mechanics and Control, 3rd Edition, Pearson Prentice Hall, Upper Saddle River, NJ.

6. R.L. Williams II, 1994, "Local Performance Optimization for a Class of Redundant Eight-Degree-of-Freedom Manipulators", NASA Technical Paper 3417.

7. S.R. Buss and J.S. Kim, 2004, "Selectively Damped Least Squares for Inverse Kinematics", typeset manuscript, April.

8. D. Tolani, A. Goswami, and N.I. Badler, 2000, "Real-Time Inverse Kinematics Techniques for Anthropomorphic Limbs", Graphical Models 62: 353–388.

9. M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M.-P. Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, and P. Volino, 2005, "Collision Detection for Deformable Objects", The Eurographics Association and Blackwell Publishing.

10. C. DeCoro and S. Rusinkiewiczy, 2005, "Pose-Independent Simplification of Articulated Meshes", ACM Symposium on Interactive 3D Graphics and Games.

11. J. Lander, 1998, "Skin Them Bones: Game Programming for the Web Generation", Game Developer Magazine, May: 11-16.

12. D.L. James and C.D. Twigg, 2005, "Skinning Mesh Animations", ACM Transactions on Graphics (SIGGRAPH 2005): 24(3), August.

13. J. P. Lewis, M. Cordner, and N. Fong, 2000, "Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation", International Conference on Computer Graphics and Interactive Techniques, 165-172.

14. A.I. Kapandji, 1974, The Physiology of The Joints volume 2, Churchill Livingstone.

15. A.I. Kapandji, 1974, The Physiology of The Joints volume 3, Churchill Livingstone.

16. A.A. White and M.M. Panjabi, 1990, Clinical Biomechanics of the Spine, Lippincott Williams & Wilkins.

CONTACT

Robert L. Williams II, Ph.D., is the corresponding author and professor of mechanical engineering in the Ohio University Russ College of Engineering & Technology; he is also a member of the Interdisciplinary Institute for Neuromusculoskeletal Research at Ohio University.

williar4@ohio.edu

http://www.ent.ohiou.edu/~bobw/html/VHB/VHB.html