A Haptic-Enabled Multimodal Interface for the Planning of Hip Arthroplasty

Nikolaos G. Tsagarakis, John O. Gray, and Darwin G. Caldwell
University of Salford, UK

Cinzia Zannoni and Marco Petrone
Biocomputing Competence Center (CINECA)

Debora Testi and Marco Viceconti
Institute of Orthopedics Rizzoli

Multimodal environments help fuse a diverse range of sensory modalities, which is particularly important when integrating the complex data involved in surgical preoperative planning. The authors apply a multimodal interface for preoperative planning of hip arthroplasty with a user interface that integrates immersive stereo displays and haptic modalities. This article overviews this multimodal application framework and discusses the benefits of incorporating the haptic modality in this area.

Multimodal environments seek to create computational scenarios that fuse sensory data (sight, sound, touch, and perhaps smell) to form an advanced, realistic, and intuitive user interface. This can be particularly compelling in medical applications, where surgeons use a range of sensory motor cues.[1-4] Sample applications include simulators, education and training, surgical planning, and scientifically analyzing and evaluating new procedures.

Developing such a multimodal environment is a complex task involving integrating numerous algorithms and technologies. Increasingly, researchers are developing open source libraries and toolkits applicable to this field such as the Visualization Tool Kit (VTK) for visualization, the Insight Toolkit (ITK) for segmentation and registration, and the Numerical Library (VNL) for numerical algorithms. Single libraries from these toolkits form a good starting point for efficiently developing a complex application. However, this usually requires extending the core implementation with new library modules. In addition, integrating new modules can quickly become confusing in the absence of a good software architecture.

To address this, researchers have developed semicomplete application frameworks that can run independently, hiding the core implementation's complexity. As such, they can be dedicated to produce custom applications.[5] However, these systems form frameworks that aren't multimodal because they don't let us integrate different visual representations or other modalities such as haptics and speech. This has motivated research in developing truly multimodal frameworks,[6] but the benefits of such integration are still largely unexplored. For the haptic modality in particular, hardware and software that can provide effective touch feedback can enhance the growth of innovative medical applications.

From this rationale, the Multisense project aims to combine different sensory devices (haptics, speech, visualization, and tracking) in a unique virtual reality environment for orthopedic surgery. We developed the Multisense demonstrator on top of a multimodal application framework (MAF)[7] that supports multimodal visualization, interaction, and improved synchronization of multiple cues.

This article focuses on applying this multimodal interaction environment to total hip replacement (THR) surgery and, in particular, to the preoperative planning surgical-access phase.[8] After validation, this approach will be highly relevant to other orthopedic and medical applications.

Hip arthroplasty planner

Hip arthroplasty is a procedure in which diseased hip joints are removed and replaced with artificial parts—the socket and prosthesis. Researchers have developed different systems for THR preoperative planning,[4,9] operating in 2D using a mouse and flat screen to produce pseudo-3D interaction. This approach makes little or no use of multisensory inputs, which leads to problems because the graphics interface strongly affects implant positioning accuracy.[10]

A team of orthopedic surgeons defined four specific tasks that form the basis for our multimodal hip arthroplasty planning environment:

- *Preparing the subject-specific musculoskeletal model.* Effective planning requires a complete

1070-986X/06/$20.00 © 2006 IEEE

and accurate musculoskeletal model usually only available from magnetic resonance imaging (MRI) data. Related work shows how we can map patient computerized tomography (CT) scans to data and models from the Visual Human to provide complete bone and soft tissue models of the hip and thigh muscles.[10]

- *Surgical-access planning.* The critical surgical-access phase consists of three main surgical tasks: determining the initial incision location and size, retracting the muscles, and dislocating the femur.

- *Components positioning.* Here the surgeon positions the prosthesis with respect to the femur. During this process, the surgeon can check functional indicators: feasibility of the planned position, primary component stability, and range of joint motion.

- *Surgical simulation.* After determining the prostheses' pose, the surgeon can interactively position the neck resection plane to verify the reamer's insertion path. Once the surgeon accepts that position, the system generates a model of the postoperative anatomy for final verifications and inspections.

The medical users exploited these surgical activities to identify the possible benefits that can be gained on these tasks by integrating the haptic modality in the preoperative planning application. Based on this study, we defined the haptic requirements of this specific application.

Haptic requirements

From this series of procedures, the medical users selected scenarios in which they felt haptic feedback would be of the greatest benefit. These included the ability to locate and size the incision, evaluate the surgical access they can achieve through that incision, and identify the functional impairment produced by any damage to the soft tissues (muscle or skin).

In addition, haptic feedback can help position and orient the implant while preventing the surgeon from positioning the component in a non-feasible location. Based on the position and orientation selected, the surgeon can evaluate this specific location using a number of haptic-enabled indicators including the thigh joint's range of motion after the simulation and the component's stability. The benefits will include accurately positioning the implant and improved execution time.

Considering these surgical activities, the medical users defined the following haptic tasks:

- *Force feedback for evaluating surgical access.* Force (or touch) feedback can help surgeons accurately locate and size an incision. During the retraction, it can help surgeons estimate the relationship between visibility and muscle damage. Force feedback can also help them evaluate the incision aperture size while dislocating the femur.

- *Force feedback for evaluating the planned-position feasibility.* Reaction forces generated by contact with the surrounding tissues let the user refine the planned position, check the feasibility of planned position, and evaluate the component's primary stability in this position.

We identified the multimodal interface's requirements using the characteristics of these haptic tasks. These requirements let us determine the necessary features of the multimodal system's software and hardware modules.

Multimodal system requirements

Any multimodal system must interact with complex data incorporating several features:

- integration of multiple I/O devices and modalities;

- seamless synchronization of the different update loops running at much different rates;

- a distributed architecture that copes with the computational load and simulation loops;

- support for complex multimodal visualization with multiple representation of the data;

- support for dynamically exchangeable haptic rendering algorithms; and

- modularity and extensibility, with simple application-level modules hiding the system architecture's complexity and synchronization problems.

We developed a multimodal application framework to address these requirements and a suitable

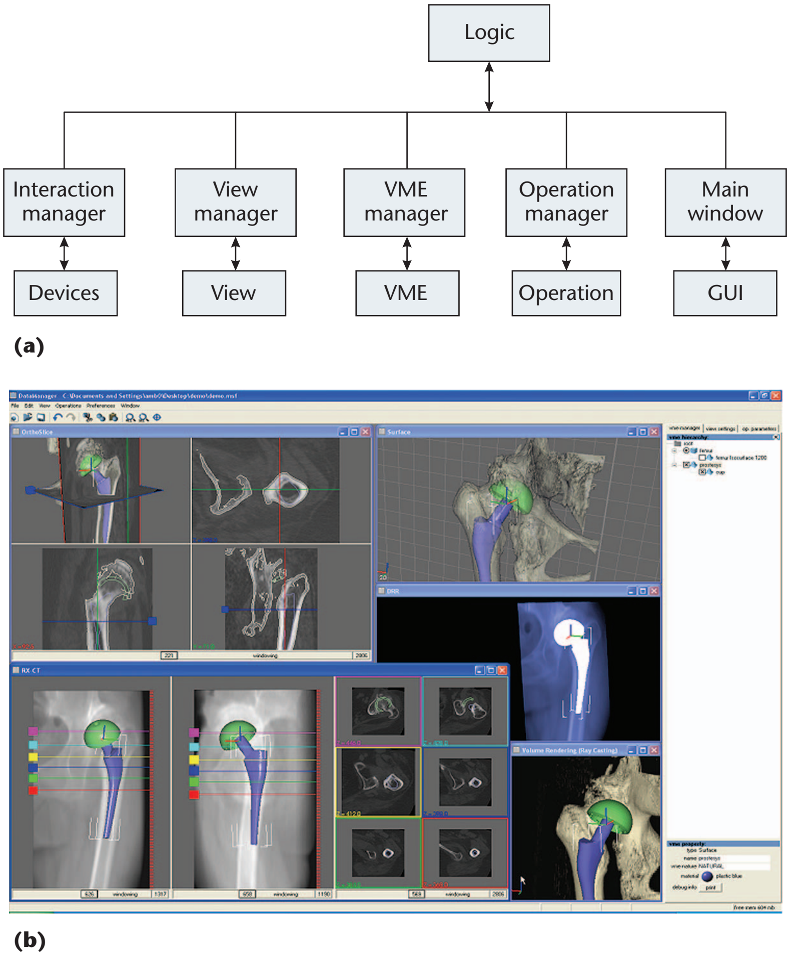


Figure 1. Multimodal application framework: (a) architecture diagram and (b) multiple display paradigm. (VME = Virtual Medical Entity.)

haptic software and hardware device to provide the haptic modltity within the framework.

Multimodal application framework

The multimodal application framework (MAF) is a software library for rapidly developing innovative multimodal environments for medical applications. It supports the Multimodal Display and Interaction (MDI) paradigm with multimodal visualization and interaction, haptics, and synchronization of the multiple cues. An MAF consists of components that control system resources, which are organized as data entities and application services. A data entity is a Virtual Medical Entity (VME). We distinguish the application services views, operations (Op), GUIs, and devices.

Every MAF application is an instance of a logic component. The logic component's main role is to control communication. Figure 1a shows the MAF architecture with the logic, manager, and all MAF resources. Figure 1b gives an example of the MAF multidisplay paradigm we used in this application.

Interaction and synchronization model

User interaction involves the I/O devices, views subsystem, and operation subsystem. The MDI paradigm requires gathering, synchronizing, and integrating inputs coming from multiple I/O devices. When users interact with the application, a stream of events is sent to the framework: discrete events (low-frequency events causing a change in the application state) and continuous events (high-frequency user interactions).

Handling input events is complex because the user might perform any set of interactive and dynamic actions. Thus, managing the interaction with a single, monolithic component is impractical. MAF involves collaboration among many components. GUI events are processed directly by application components (for example, operations or logic), and events coming from I/O devices are typically processed by specific objects named *interactors*. The general MAF interaction model for I/O devices implies three elements: a semiotic unit (I/O device), semantic unit (interactor), and an application component.

MAF manages interactions with multiple I/O devices, through routing, locking, and fusion mechanisms within the Interaction Manager. This synchronizes inputs from different devices using the Device Manager subcomponent responsible for keeping the list of connected devices and for synchronizing their inputs with the application's main visualization loop (see Figure 2). For haptic devices, which require high and decoupled update rates, high-speed loops run inside the haptic subsystem, and only events sent at visualization rates pass to and from the MAF. MAF ensures synchronization by sending events to them—for example, each time a haptic surface rendering is started an event containing the rendered surface is sent to the haptic device and, hence, to the haptic rendering server/library. This data synchronization is rare, so it has minimal overhead.

During continuous interaction, the visualization and haptic loops are synchronized by the haptic device sending events (at the graphics rate) to the visualization loop. Hence, we can compute haptic and graphical models in a decoupled but synchronized fashion.

System hardware architecture

To address the intensive computation needs and particularly to accommodate the different update rates (for example, visualization systems update at 50 to 100 Hz while haptic devices

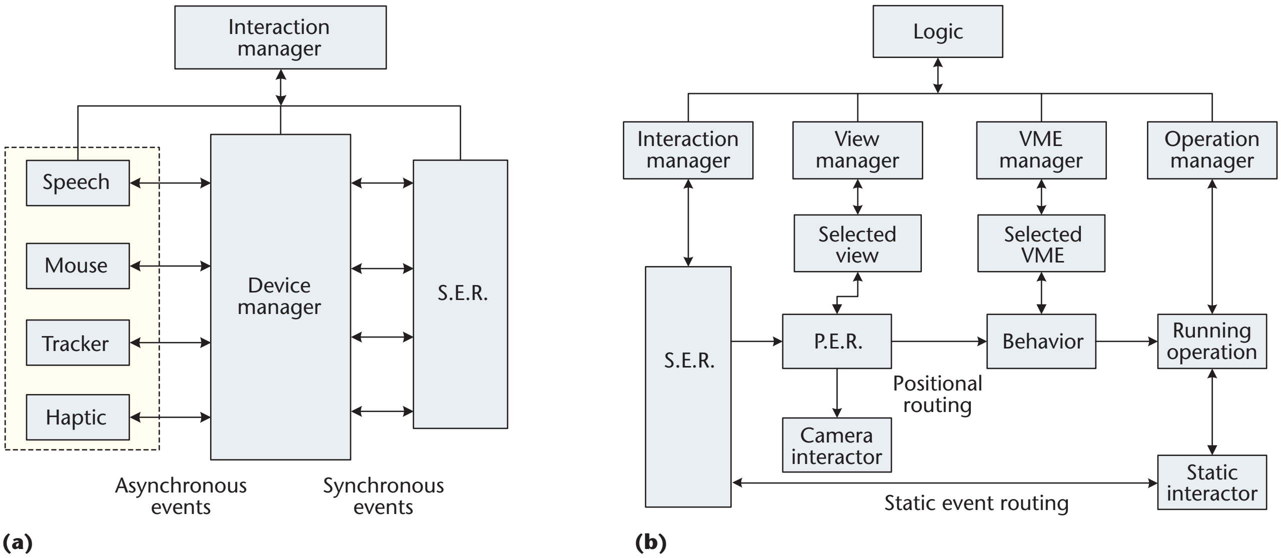


Figure 2. (a) Synchronizing the device's events, and (b) multimodal application framework (MAF) static and positional event routing (S.E.R. and P.E.R.).

update at more than 1,000 Hz), the multimodal system architecture uses multiple networked dedicated machines (see Figure 3).

We use a dedicated graphics server to perform the graphics rendering, and the haptic server manages the haptic subsystem via a TCP/IP interface. The advantage of this approach over a single machine, multithreaded approach is that it minimizes the coupling between local rates running on different machines. Also, the rates for critical processes such as the haptic servo input and feedback control process are more consistent, enabling stable haptic rendering even for complex environments. This approach also gives us separate, extensible, and reconfigurable control of the different input feedback subsystems. The disadvantage is it increases synchronization requirements between the various subsystems, which the MAF directly addresses.

Haptic subsystem implementation

We designed a haptic device to support either one- or two-handed operation and fabricated it to suit the application workspace and interaction requirements we defined earlier. The device consists of two three-degrees-of-freedom (DOF) closed-chain mechanisms, each forming a classic five-bar planar device that can also be rotated around the axis along its base (see Figure 4a, next page). We selected the inner and outer link lengths to provide a workspace that satisfies the motion-range requirements of both the surgical-access and the component-position-feasibility tasks.

To support two-handed interactions, we can configure the device to work in two modes. The *double-independent mode* provides two mechanisms (6-DOF input and 3-DOF feedback) with two separate haptic interface points; the *coupled mode* configuration provides a single linked

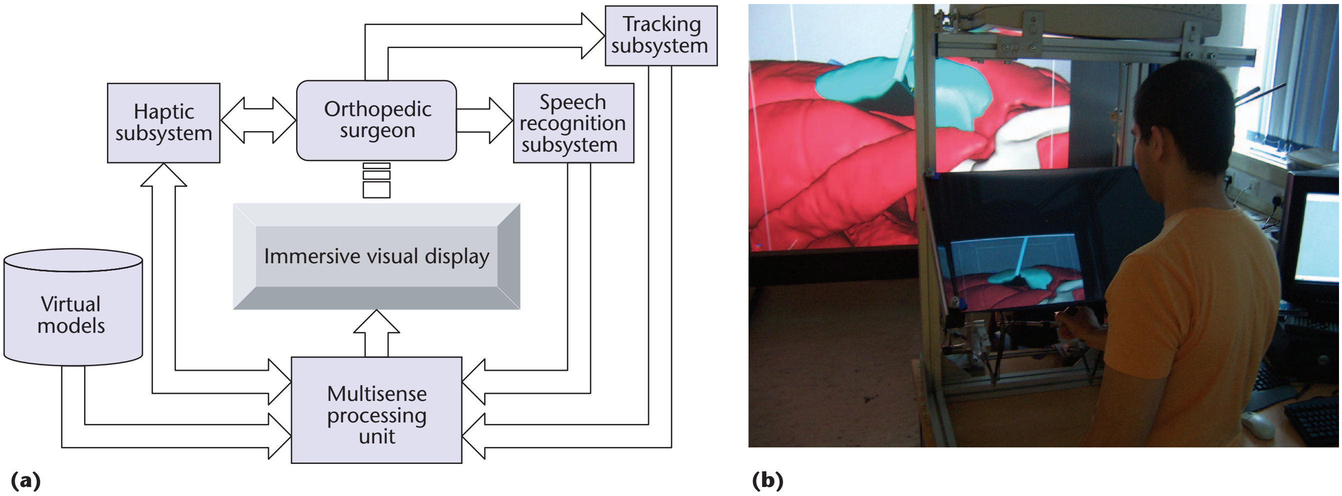


Figure 3. (a) Multimodal system architecture and (b) the system hardware setup showing the integration of the immersive stereo display and the haptic device.

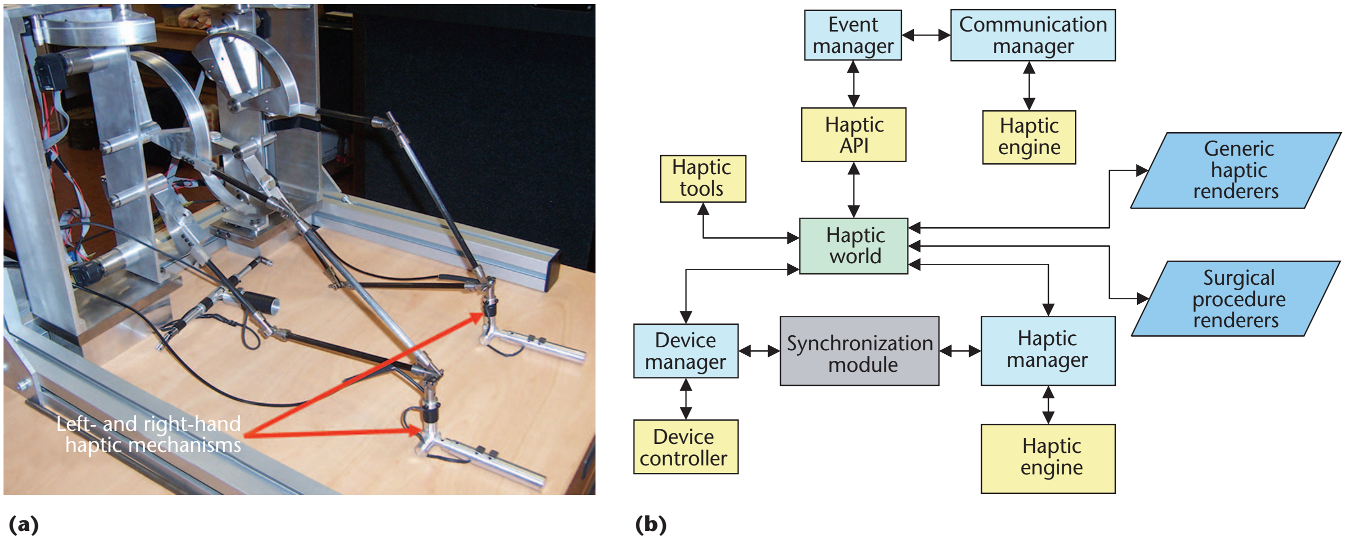


Figure 4. (a) Prototype haptic desktop device. (b) Haptic rendering library architecture and interaction among the library modules.

mechanism (6-DOF input and 5-DOF feedback).[11]

The haptic rendering library coordinates the input and haptic feedback signals. We developed this library to support the haptic modality within the MAF (see Figure 4b). We use a multithreaded approach that includes four core processes: device control, haptic rendering, event and command handling, and communication. Four respective managers manage these process threads.

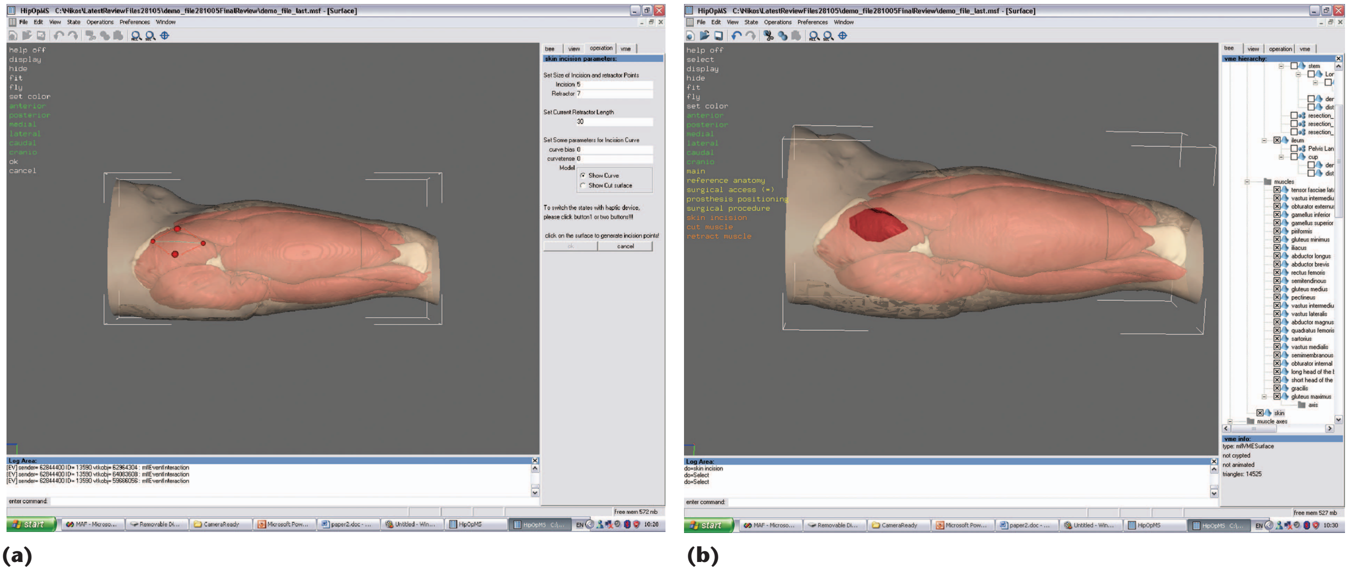
We provide a haptic tool as a mechanism for force feedback and couple it to the haptic device within the haptic manager object. The haptic rendering process, managed by the haptic manager, uses the current tool to gather force requests within the haptic world space and asks the device to supply the user with the computed haptic feedback. The haptic subsystem runs on a dedicated haptic machine, and communication between the haptic module and the visualization station is performed using the haptic subsystem API.

Surgical-access haptic modules

Surgical-access planning consists of a skin incision, muscle retraction, and femur head dislocation. The initial incision is defined by two reference points under force-feedback control. With the incision defined, the surgeon controls the aperture size using two additional reference points automatically created when the incision line is defined (see Figure 5).

We implemented the incision haptic renderer as a standard surface renderer, letting surgeons accurately locate the reference points while providing them with feedback on the constraints imposed by the skin surface.

Figure 5. (a) The surgeon selects the skin incision size and (b) views the incision aperture.


skin incision parameters:

Set Size of Incision and retractor Points

Incision 5

Retractor 7

Set Current Retractor Length

30

Set Some parameters for Incision Curve

curve bias 0

curvetense 0

Model ● Show Curve ○ Show Cut surface

To switch the states with haptic device, please click button1 or two buttons!!!

click on the surface to generate incision points!

```
Log Area:
[EV] sender= 62844400 ID= 13590 vtkobj= 62964304 : mflEventInteraction
[EV] sender= 62844400 ID= 13590 vtkobj= 64083608 : mflEventInteraction
[EV] sender= 62844400 ID= 13590 vtkobj= 59666056 : mflEventInteraction
```

```
Log Area:
do=skin incision
do=Select
do=Select
```

vme info:
type: mflVMESurface
not crypted
not animated
triangles: 14525

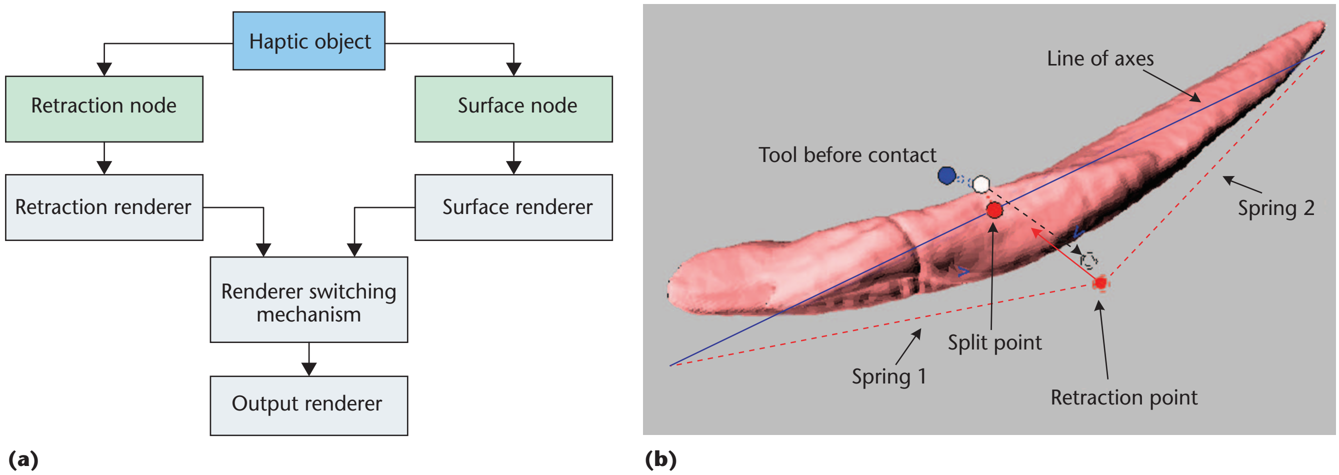


Figure 6. (a) The muscle retraction haptic object with the integration of the surface and retraction haptic node, and (b) the two-spring retraction haptic render showing the two spring elements' initiation and termination points.

The skin incision and retraction is followed by the much more complex task of muscle retraction, which has a higher probability of damaging muscles and other tissues. During this procedure, the surgeon introduces the retractor between the muscles and retracts one toward the edge of the skin incision. The retracted muscle is held in position while the next muscle is retracted and so forth until the head of the femur and the acetabulum are visible.

To simulate this, the haptic and visual subsystems must cooperate within the MAF to provide the correct level of synchronization. We implemented the muscle retraction haptic renderer, which lets the user estimate the trade-off between visibility and muscle damage during the retraction, as a combined haptic node formed from two haptic renderers (see Figure 6a). The first node represents the surface of the muscle to be retracted and is implemented as a surface haptic renderer permitting interaction with the muscle surface. The second node implements the retraction haptic model realized using a two-spring (200 to 400 Newton meter [N/m]) model (see Figure 6b).

We tuned the spring parameters using a Finite Element (FE) analysis of muscle and actual patient data. The state of the MAF operation controls switching between the surface and retraction models. When the femur and acetabulum are visible, the femur head can be dislocated from the socket to allow access through the aperture (see Figure 7). To let the user assess the difficulty in operating through the aperture, we generate force feedback during the dislocation.

This is modeled when a collision occurs between the femur head and the surrounding soft tissue objects (muscles and skin). To simulate the resistance caused by muscle elongation, we generate additional feedback forces from the muscles connecting the femur and ileum. We modeled each muscle as a spring/damper:

$$\mathbf{F}_m = K_M \cdot d_m \cdot \mathbf{a}_m + B_m \cdot \mathbf{u}_f \times \mathbf{a}_m$$

$$\mathbf{F}_{elongation} = \sum_{i=0}^{N} {}^{i}\mathbf{F}_m \qquad (1)$$

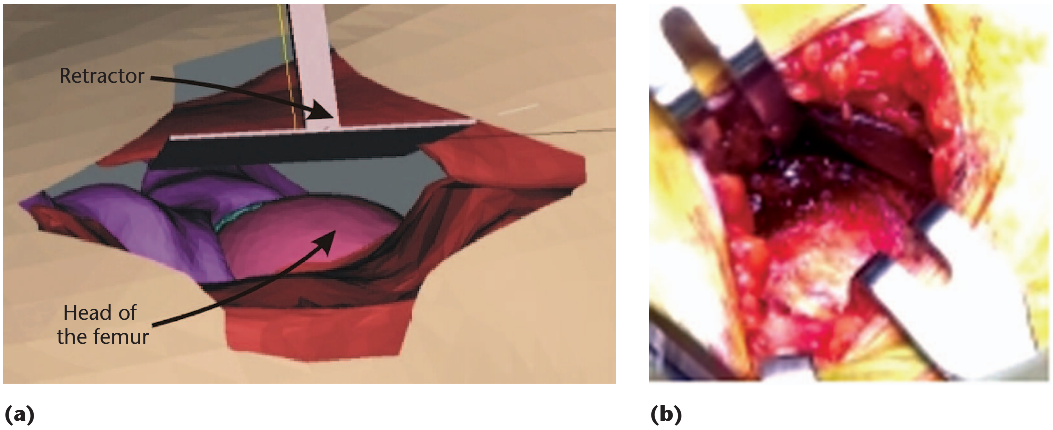


Figure 7. (a) A visualization of the muscle retraction, and (b) an example of surgical access with a retracted muscle.

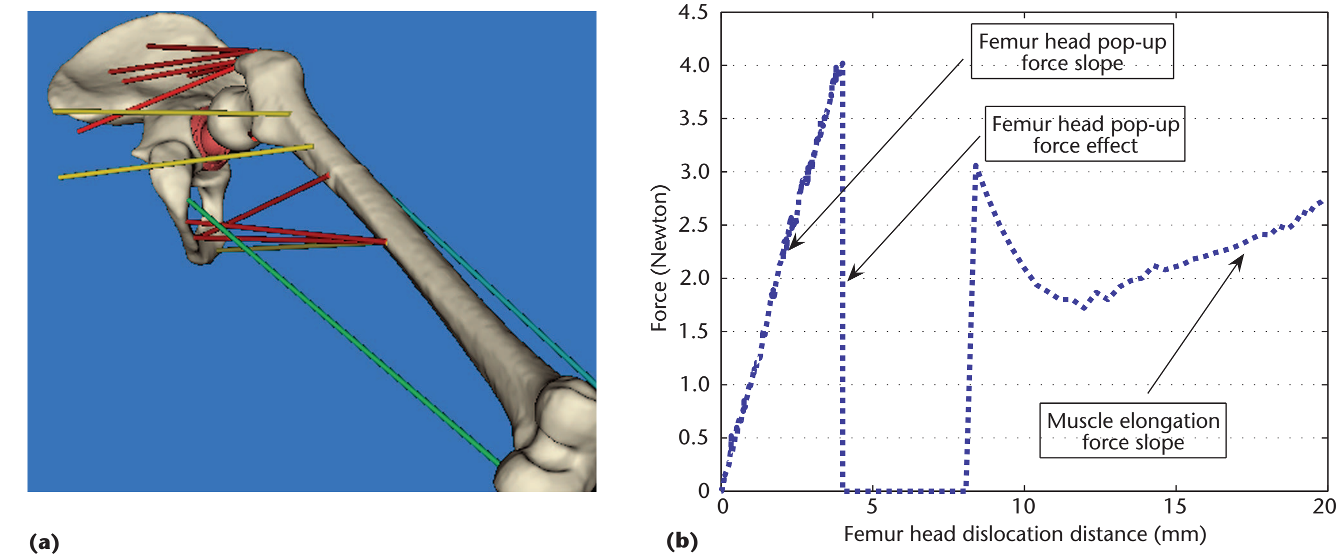


Figure 8. (a) A visualization of a femur dislocation, and (b) the force profile recorded during the dislocation.

where K_M and B_m are the specific muscle stiffness and damping parameters, d_m is the muscle elongation, $\mathbf{a}_m$ is the unit vector of the muscle line of axes, and $\mathbf{u}_f$ is the femur bone velocity vector. This is adequate because the rendering of the force generated by the muscle elongations is a simulation feature we added to improve realism in this operation and doesn't affect the actual planning. To permit interactive operation during the dislocation, the system visualizes the muscle lines of axis with coloration dependent on the strain (see Figure 8a).

To improve the realism of a femur head dislocation, we implemented a pop-up effect by connecting a strong (1000 N/m) spring—only active in close proximity to the socket's center, between the femur head and socket centers:

$$\mathbf{F}_{popup} = \left[K_E\right]\cdot\, \mathbf{p}_d + \left[B_E\right]\cdot\, \mathbf{u}_f, \left|\mathbf{p}_d\right| < r_d$$

$$\mathbf{F}_{popup} = 0, \left|\mathbf{p}_d\right| > r_d \qquad (2)$$

where K_E and B_E are the pop-up stiffness and damping parameters, $\mathbf{p}_d$ is the femur head dislocation distance vector, r_d is the pop-up spring's active sphere radius, and $\mathbf{u}_f$ is the femur bone velocity vector. When the femur head dislocation distance becomes greater than this distance, a force discontinuity is created dropping the force to 0 Newton (N) for 40 milliseconds (see Figure 8b), creating a discontinuity that the user perceives as the femur head popping out.

Preliminary experimental results

We performed two preliminary validation experiments involving five subjects to evaluate the multimodal benefits and effectiveness using the system shown in Figure 3. Two subjects involved in the system development were well-trained users, and three subjects had no previous system experience.

We gave each user an explanatory test sheet. Experiment 1 evaluated the benefits of force feedback on the accuracy of defining the incision aperture. A reference aperture was defined, and the subjects were asked to execute a skin incision using the immersive multimodal interface with the haptic feedback modality active in the first case. Each subject tried to replicate the reference aperture. The users then repeated the process with the haptic modality disabled.

In the second case, with the haptic modality disabled, the system provided visual feedback (color changes) indicating contact between the device avatar and the skin. The users repeated the test five times for each case. We recorded the time required to carry out the positioning. Figure 9 gives the distance error between the position of the incision points and the position of the points of the reference incision with the haptic multimodal interface enabled and disabled.

The users obtained significantly higher accuracy using force feedback in this experiment. Their execution time was also considerably reduced. The average execution times we recorded for the haptic- and nonhaptic-enabled execution were 31 and 57 seconds, respectively. This shows that integrating the haptic cues provides benefits in terms of accuracy and execution time.

Another important observation is there was no significant difference in the performance among the five subjects. This initial indication

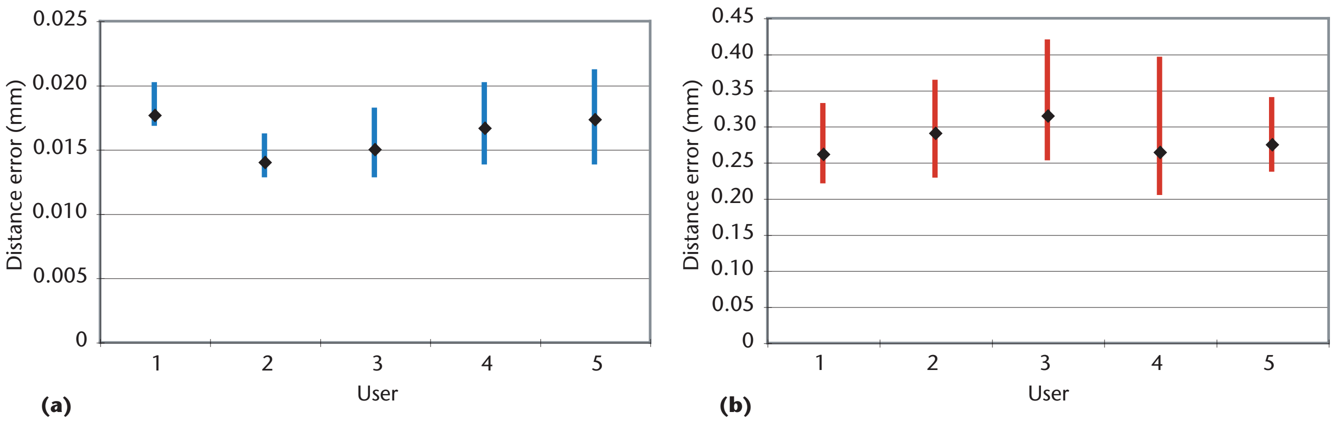


Figure 9. Experimental results for distance error with (a) the haptic modality enabled and (b) the force feedback disabled.

shows that the user effect wasn't significant in the multimodal environment, and we achieved a good level of accuracy and usability even with minimal prior experience.

Experiment 2 demonstrated the benefits in terms of execution time using the two-handed interaction paradigm. We configured the haptic device to work in the two-handed operation mode where the left part of the device was for tool manipulation and the right for manipulating the camera of the visual scene. We asked the subjects to position and orient the retractor tool close to a predefined location between two muscles without actually performing the retraction. This task required complex manipulations of the retractor tool while simultaneously manipulating the camera. The subjects performed this operation five times using the haptic device, and then they repeated the same process using a standard mouse.

We recorded the execution times in both cases. The time required to execute the task using a two-handed interaction was considerably reduced (on average 43 seconds) compared to that achieved with the mouse (on average 105 seconds). These results show the effectiveness of this type of interface when complex manipulation is necessary.

Conclusions

We're currently evaluating the complete multimodal system. Our initial experiments have helped us validate the multimodal interface in the surgical access task. We might also see benefits from using this multimodal interface in other aspects of medical planning. We plan to extensively evaluate these avenues in the future to assess the usefulness of the planning procedures' various modules. These tests will involve a broader selection of clinical users.

We'll also work on enhancing the system's ability to address the haptic tasks requirements of other surgery procedures. These will include alterations or trimmings in both the system hardware (mechanical structures) and software with the incorporation of other surgical haptic renderers to form a library of surgical haptic procedures. **MM**

Acknowledgments

This work was supported by the Multisense European project (IST-2001-34121).

References

1. H.D. Dobson et al., "Virtual Reality: New Method of Teaching Anorectal and Pelvic Floor Anatomy," *Dis Colon Rectum*, vol. 46, no. 3, 2003, pp. 349-352.
2. M.A. Spicer and M.L. Apuzzo, "Virtual Reality Surgery: Neurosurgery and the Contemporary Landscape," *Neurosurgery*, vol. 52, no. 3, 2003, pp. 489-497.
3. S. Hassfeld and J. Muhling, "Computer Assisted Oral and Maxillofacial Surgery: A Review and an Assessment of Technology," *Int'l J. Oral Maxillofacial Surgery*, vol. 30, no. 1, 2001, pp. 2-13.
4. H. Handels et al., "Computer-Assisted Planning and Simulation of Hip Operations Using Virtual Three-Dimensional Models," *Studies in Health Technology and Informatics*, vol. 68, 1999, pp. 686-689.
5. M. Fayad and D.C. Schimidt, "Object-Oriented Application Frameworks," *Comm. ACM*, vol. 40, no. 10, 1997, pp. 32-38.
6. F. Flippo, A. Krebs, and I. Marsic, "A Framework for Rapid Development of Multimodal Interfaces," *Proc. 5th Int'l Conf. Multimodal interface*, ACM Press, 2003, pp. 109-116.
7. M. Krokos et al., "Real-Time Visualisation within the Multimodal Application Framework," *Proc. 8th Int'l Conf. Information Visualization* (IV04), ACM Press, pp. 21-26, 2004.

8. C.G. Schizas, B. Parker, and P.-F. Leyvraz, “A Study of Pre-Operative Planning in CLS Total Hip Arthroplasty,” *Hip Int'l*, vol. 6, 1996, pp. 75-81.
9. S. Nishihara et al., “Comparison of the Fit and Fill between the Anatomic Hip Femoral Component and the VerSys Taper Femoral Component Using Virtual Implantation on the ORTHODOC Workstation,” *J. Orthopaedic Science*, vol. 8, no. 3., 2003, pp. 352-360.
10. M. Viceconti et al., “CT-Based Surgical Planning Software Improves the Accuracy of THR Preoperative Planning,” *Medical Eng. & Physics*, vol. 25, no. 5, 2003, pp. 371-377.
11. N.G. Tsagarakis and Darwin G. Caldwell, “Pre-Operative Planning for Total Hip Replacement Using a Desktop Haptic Interface”, *Proc. IMAACA 2004*, DIP Univ. of Genoa, 2005, pp. 209-216 .

Nikolaos G. Tsagarakis is a research fellow at the University of Salford, UK. He works in rehabilitation, medical robotics, and haptic systems. Other research interests include novel actuators, dextrous hands, tactile sensing, and humanoid robots. Tsagarakis received his PhD in robotics and haptic technology from the University of Salford. He is a member of the IEEE Robotics and Automation Society.

Marco Petrone is a staff member with the High Performance Systems, Visualization Group (VISIT) at the Biocomputing Competence Center (CINECA). His research interests include scientific visualization, biomedical applications, multimodal interaction, and the multimodal application framework (openMAF). Petrone received a degree in computer engineering from the University of Padova, Italy.

Debora Testi is a researcher at the Laboratorio di Tecnologia Medica of the Institute of Orthopedics, Rizzoli. Her research interests include bone remodeling, osteoporosis, femoral neck fractures, and software for computer-aided medicine. Testi has a PhD in bioengineering from the University of Bologna. She is a member of the European Society of Biomechanics.

Cinzia Zannoni is a project manager with the High Performance Systems group at the Biocomputing Competence Center (CINECA) and is the coordinator of VISIT, which runs activities in scientific visualization and the development of IT services for the support of scientific communities. Zannoni received a PhD in bioengineering at the University of Bologna.

John O. Gray is a professor of advanced robotics at the University of Salford. His research interests include medical robotics, nonlinear control systems, precision electromagnetic instrumentation, and robotic systems for the food industry. Gray received a PhD from the University of Manchester in control engineering.

Marco G. Viceconti is the technical director of the Laboratorio di Tecnologia Medica of the Institute of Orthopedics, Rizzoli. His research interests are in developing and validating medical technology for orthopedics and traumatology. Viceconti received a PhD in bioengineering from the University of Florence. He is currently the secretary general of the European Society of Biomechanics and a member of the Council of the European Alliance for Medical and Biological Engineering and Science (EAMBES).

Darwin G. Caldwell is the Chair of Advanced Robotics in the Center for Robotics and Automation at the University of Salford. His research interests include innovative actuators and sensors, haptic feedback, force-augmentation exoskeletons, dexterous manipulators, humanoid robotics, biomimetic systems, rehabilitation robotics, and robotic systems for the food industry. Caldwell received a PhD in robotics from the University of Hull. He is chair of the United Kingdom and Republic of Ireland (UKRI) region of the IEEE Robotics and Automation Society.

Readers may contact Nikolaos Tsagarakis at the School of Computer Science and Engineering, University of Salford; n.tsagarakis@salford.ac.uk.