Simulator development in vascular and visceral interventions

Derek A. Gould, Fernando Bello, Nigel John, Sheena Johnson, Carrie Hunt, Helen Woolnough, Andy Bulpitt, Vincent Luboz, Daniel King, Pierre-Frederic Villard, et al.

2

SIMULATOR DEVELOPMENT IN VISCERAL AND VASCULAR INTERVENTIONS

DEREK GOULD, FERNANDO BELLO, NIGEL JOHN, SHEENA JOHNSON
CARIANNE HUNT, HELEN WOOLNOUGH, ANDREW BULPITT
VINCENT LUBOZ, DANIEL KING, PIERRE-FREDERIC VILLARD
SANDHYA PISHARODY, FRANCK VIDAL, AMRITA SINHA

Throughout the practice of procedural medicine, there is an unrelenting shift to management by less invasive techniques such as interventional radiology (IR). This subspecialty within radiology uses imaging to guide needles, wires and catheters using tiny access incisions. Like other minimally invasive techniques, risk, pain and recovery times are reduced as compared with more invasive approaches such as open surgery. These benefits, alongside the emergence of increasingly novel therapeutic technologies, are driving worldwide expansion.

The core skills of IR include the Seldinger technique and the use of imaging and touch to effectively guide needles, wires and catheters in a wide range of procedures. Safe practice requires the operator to respond correctly to both visual and tactile cues in vascular angioplasty, stenting and stent-grafting, as well as control of bleeding, biopsy, abscess drainage and catheterization of the urinary and biliary tracts for drainage and stenting. The operator's deliberations may then initiate and inform a range of motor actions, including very fine translational and rotational motions, particularly in challenging anatomy. As the spectrum of available techniques increases, so the limited number and availability of suitably trained practitioners becomes a factor in their restricted availability to patients. Awareness of the need to expand IR training facilities is thus highly topical.

IR training: problems and solutions

IR

For nearly half a century an apprenticeship in patients, with expert supervision, has served to train these core skills, commencing with straightforward, diagnostic angiography, then progressing through more complex cases. There are, however, many drawbacks to this training, including the costs and time for supervision and the inevitable added risks and discomfort for patients. At the same time, diagnostic angiography is now rapidly being replaced by non invasive imaging methods (computed tomography [CT] and magnetic resonance [MR]), reducing the opportunities to acquire basic clinical skills. Pressures to improve throughput in the national health system, together with the Calman system in the UK, and implementation of European and US Working Time Directives are increasing the difficulty for trainees to acquire experience in a time efficient manner. Popularity of interventional training impacts on the available case mix, which may vary between centers, and inevitably limits exposure to rare and adverse events (complications, allergy, anaphylaxis). Paradoxically, with increasing complexity of interventional procedures, the risk of learning fine motor skills in patients increases. Finally, there is a need to consider the limitations of the assessment tools available to today's mentor: in the UK, certification of IR skills still uses a logbook method indicating the number of procedures performed. Clearly not only do we need new methods of interventional skills training, there also needs to be a paradigm shift to more objective methods of assessment such as exist in surgery.

New skills training paradigms

Models can be created using rapid prototyping though these are expensive, lack physiology and 'feel', and their anatomical detail is difficult to alter. When training needle puncture, models lack robustness and are destroyed by repeated use. Animal models incorporate physiology but with anatomical differences, a lack of pathology and, in the UK and USA, political acceptability. Perhaps most topically is the use of computers and human-computer interface devices to replicate the visual, spatial and tactile characteristics of procedures in patients. Thus computer simulations provide a means to address shortcomings in other simulator technologies, using virtual environments developed from medical imaging data [1]. In this way, computer simulation could, in principle, mirror a task in the real world so effectively that an observer suspends their disbelief.

The operator's experience of a simulation depends on its content (what we can actually do in the simulation), appearance and 'feel', as well as the level of fidelity (faithfulness of replication) to which all this is modelled. Yet after a first flush of excitement in the 1990s, serious development of 'virtual reality' simulations slipped from the academic and commercial agenda when the limitations of available technology left this a seemingly unattainable ideal. Modern graphics cards, imaginative interface devices and the greater fidelity potential of more finely structured virtual environments have now directed serious interest towards correctly replicating the finer motor skills that are important to perform safe, effective minimally invasive techniques. For this to occur, key steps in a simulation are first identified by a team of human factors experts using cognitive task analysis (CTA), or task analysis (TA): this then acts as the basis of simulator development. Without this there would be little hope to mirror the real world task or provide effective and valid training; akin to building a house without an architect's plan! This 'content' of a simulation is key to its utility for training and assessment; elements that are correctly modeled can be used to train in a curriculum, provided other means are used to train steps and tasks that are missing from the model. If this is not the case, a misguided belief in the efficacy of a simulation could jeopardize the safety of future patients.

We will describe an approach that aims to meet these requirements, to provide solutions to some novel interventional training needs such as the Seldinger technique and the use of ultrasound for guidance. Indeed recommendations now exist for using ultrasound imaging to guide central venous catheter insertion. Similar advice for the placement of chest drains follows reports of 12 deaths and 15 incidents of serious harm in this procedure over three years from January 2005 to March 2008. Several other less severe cases were thought likely to have gone unreported [2]. It is clear that the safety of such techniques will mandate their use throughout mainstream medical practice; yet they require correct training, the solution to which lies little further than the PC on your desk.

Simulation: roles and technologies

The input data to a simulator is usually patient specific medical scan data such as CT, MRI or other imaging modality, in the digital imaging and communications in medicine (DICOM) format. Stock anatomical models may also be used, plus computer-aided design (CAD) models of any instruments or other equipment needed. The computer processor has much work to do. Segmentation, volume and/or surface rendering, collision detection, soft body deformation, cutting and other effects must be carried out, often in real time. The user interacts with the three-dimensional (3-D) models in the simulator using haptics, stereoscopy, audio, etc. Often a mannequin will also be integrated into the environment, although this does limit the variations possible in the simulation. The following sections discuss many of the different options available for building a medical procedures training simulator.

Data sources

Notwithstanding the requirements for patient privacy and ethical constraints, access to patient image data has never been more straightforward. The DICOM standard has ensured interoperability between medical scanners and the hospital databases and picture archiving and communication systems (PACS). Many hospitals today also provide secure access to image data over the World Wide Web or via a bespoke telemedicine application. Whereas maximum flexibility can be provided within a medical simulator if patient specific data is used on demand, there are also many other sources of high quality data sets available for training and education purposes. The most well known is the visible human data set [3]. The detailed scans and images from this and other similar sources have been used to generate anatomical objects for many simulators.

Data descriptions of all of the instruments (needles, guidewires, scalpel, trocars) used in a particular medical procedure are also required. It is relatively straightforward to create a geometric model of an instrument/tool using your favourite CAD package. However, access to data on the physical properties of instruments is more difficult to obtain. Medical manufacturers of catheters and guidewires, for example, will not disclose such information as they regard this as their intellectual property. Also, obtaining information about the physical properties of human soft tissue is non trivial as it is a viscoelastic material with complex behaviour that varies enormously with tissue type, age, sex and other factors. A good summary of the problems of tissue modelling and characterization can be found in Liu et al. [4]. Therefore, to be able to model how much force is used when carrying out different tasks, experimental data must be acquired. There is relatively little of this type of data reported in the literature, and where it does exist the accuracy is limited as experiments are typically carried out ex-vivo, for example [5], or in-vivo using animal studies, for example [6].

Processor hardware

The computer processor is at the core of the medical simulator and has always been one of the major technological factors responsible for the advancement and assimilation of medical virtual environments. The hardware platform has to be chosen carefully so that real time performance can be maintained whilst executing complex software algorithms and dealing with all user interactions. It has been evident for some time that the immediate future lies with commodity off-the-shelf computer graphics hardware. Stemming from the demands of the gaming industry, companies such as NVIDIA (Santa Clara, California) and AMD/ATI (Sunny-Sunnyvale, California) have driven down the cost of graphics hardware whilst significantly increasing performance. Combined with simultaneous improvements in central processing unit CPU technology and the increased bandwidth of the internal communication bus, it is now possible to simulate many complex procedures fast enough for inclusion in a training simulator. The graphics processing unit (GPU) can be programmed directly providing the developer with access to sophisticated parallel processing graphics hardware, which can then be used to implement computationally expensive algorithms (not necessarily graphical algorithms). Development environments such as NVIDIA's compute unified device architecture (CUDA) are now available to the programmer to facilitate access to the GPU, and CUDA applications are already providing significant performance gains.

Algorithms and software

The simulation software has to make effective use of the hardware components described above and

also implement sophisticated techniques including 3-D segmentation of medical scan data, soft tissue modelling, collision detection, and other dynamic and physiological processes. Low level application programming interfaces (API) such as OpenGL and DirectX are commonly used for the visualization component and these continue to support the new functionality of graphics hardware. For surgical simulation and haptics, higher level APIs are becoming available. Chai3D (from Stanford University) and H3D (from SenseGraphics, Sweden) are two examples of open source haptics development environments. Also widely used is the Visualization ToolKit (VTK), which supports scalar, vector, tensor, texture, and volumetric methods; and advanced modelling techniques such as implicit modelling, polygon reduction, mesh smoothing, and cutting.

Software specifically designed for medical simulation is available as open source. The Simulation Open Framework Architecture (SOFA), SPRING, and the General Physical Simulation Interface (GiPSi), are the most well known examples. They aim to provide the developer with ready access to the well established algorithms without having to re-invent the wheel, plus provide an appropriate framework for the introduction of new techniques. For example, these APIs support the two physics-based techniques widely implemented for soft tissue deformations: mass-spring models and finite element modelling (FEM), plus algorithms for fluid flow, articulated bodies and force modelling.

Another challenge in building a medical simulator is to calculate collision detection between instruments and tissues in real time and to provide an appropriate response. For high fidelity, the exact instant of each collision needs to be calculated before updating the configuration of the physical bodies. A closed form solution only exists in simple cases, however, and so a numerical root finder is usually involved. Self intersection of tissues causes additional problems. There is much ongoing research to optimise collision detection algorithms exploiting temporal coherence, low resolution proxies, spatial partitioning, etc. GPU based implementations are also being developed (and SOFA is using CUDA to speed up collision detection). The collision response must then determine the deformation effect on soft tissues, possible cutting or tearing of the tissue, and an appropriate force and tactile feedback that can be fed into the haptics hardware.

TA and metrics: basis of simulation

The majority of skills teaching and training within health care, is based upon the traditional apprenticeship model. This is the case in IR where proficiency on a task is attained through gaining experience on patients under the mentorship of an expert. Experts are required to oversee the work of trainees and provide the training necessary for trainees to become competent practitioners. However, there are major drawbacks to this training method and there is a requirement to develop alternative methods. Neequaye et al. [7] in their review of endovascular skills training and assessment list the available alternatives as synthetic models, anesthetized animals, human cadavers and virtual reality simulation. The benefits of using simulators to train include the opportunity to gain this early experience in an environment free from risk to patients, an opportunity to learn from mistakes and the potential to rehearse complex cases prior to performing the intervention on a patient. There are also cost benefits with one simulator able to provide training opportunities to numerous trainees over an extended period of time. However, simulators can be criticized for their lack of relevance to real life situations. For example, in the atudy by Van Herzeele et al. [8], when rating the simulator for realism and training potential on a five-point Likert scale, interventional radiologists produced a median score of only 3 to 3.5 revealing that they were unsure as to the realism and value of the simulator.

TA techniques have played a critical role in the development of training and system design for the past 100 years and informed much of the health and safety legislation in operation today [9]. Although still in its infancy as a tool within healthcare, TA techniques are increasingly being used as an educational resource within the medical community and have been successfully applied in a limited number of studies. For example, Grunwald et al. [10] described the use of CTA in the development of surgical training, and Velmahos et al. [11] applied it to the teaching of technical skills within surgical skills laboratories. To the best of our knowledge however, no interventional vascular simulators have been developed through the use of a detailed TA of the real world tasks to be simulated.

TA is used to identify the individual steps that need to be performed in order to complete a given task. The structure and order of individual steps are investigated in detail and reproduced in TA documentation. A complete TA will detail step-by-step each point that needs to be conducted in order to successfully complete a task. CTA has been described as an extended form of traditional job analysis with a focus on the knowledge and thought processes that support observable task performance [12]; and as a '*set of methods for identifying cognitive skills, or mental demands, needed to perform a task proficiently*' [9]. Clark and Estes [13] detailed the significance of CTA as a training tool and described training on cognitive processes and structures as superior to behaviourally based training systems for training and development on complex tasks. CTA is therefore particularly relevant for use in deconstructing tasks which are conducted by experts and where the knowledge these experts hold and the decisions they make during the completion of a task are essential to the successful completion of a task. Further benefits of using CTA as a training tool include speed and cost reductions in a training program.

As previously detailed, it can be difficult for experts to provide trainees with all the information they need to successfully complete a task since they often omit key parts of the tasks due to the automation of their own skills. The same issues arise when asking experts to provide detailed information on a task in order that a detailed task description can be developed. The role of a task analyst researcher is therefore to ask detailed questions of the experts in order to get them to verbalise all steps included in a task. It is important that a TA researcher is a trained analyst and has little detailed knowledge of the tasks to be deconstructed. Data was collected in this study through 1) discussions with CRaIVE clinician collaborators; 2) direct observation of procedures; 3) videoing procedures for use both as an observational aid for the task analysts and as a cue during interviews with experts to facilitate the detailed discussion of particular points during a procedure; 4) interviews with subject matter experts.

The TA documents have been utilized by the computer science teams on the project to aid the development of interventional radiology simulators for training. They have also been used to facilitate the collection of metric data designed to identify key parts of the procedures. Metric data collection has been gathered via a critical procedure steps (CPS) questionnaire which was designed to gather information on objective assessment. It was anticipated that some steps would be of greater importance than others, however, in order to ensure that the correct information was gathered it was necessary to ask experts to rate each step in turn. By gathering information in these two areas we were able to identify those areas that were key to performing a successful procedure. This information is important for assessing trainee performance whilst learning these procedures. For example, if a step is rated very important and very risky this would be paid particular attention during trainee assessment. A step rated as not important and posing little risk would receive less attention (although it will not be ignored completely). The CPS questionnaire for each procedure was designed based on the information in the TA and was developed in line with discussions with project collaborators in order to ensure areas of specific importance to simulator design were included. The questionnaire covered two main areas: 1) the importance of a procedure step for completion of a successful procedure; and 2) the amount of risk a step poses to the patient. Additional questions included whether a step resulted in common trainee errors and whether or not a step required fine motor skills.

3-D anatomical modelling

The medical images used to plan and guide interventional radiological procedures come mainly from four imaging modalities: CT, MRI, rotational angiography, and ultrasound. Contrast agent can be used to enhance the visibility of certain structures. The use of a series of slices is currently the most common visualization technique, with interventional radiologists building a 3-D mental model of the anatomy [14]. However, at times, using only 2-D slices is not practical because of complex or branching structures such as blood vessels which are difficult to follow from slice to slice. In these cases, a 3-D view may prove more useful. Direct volume rendering [15] allows a visualization of the entire data set: however, it is hard to clearly identify the contours of a specific structure. This identification is done through the segmentation of the objects. A review of segmentation methods is presented below after an introduction to pre- and post-processing methods used to improve the visual

appearance of an image. The segmentation result is usually a dataset of the same dimension where the structure of interest is clearly distinguishable. Nevertheless, manipulation and simulation of anatomical structures is difficult for volumetric images because of the large amount of data involved. It might therefore be necessary to instead reconstruct the 3-D surface of the structure using a triangular mesh, which allows efficient rendering by graphics hardware and can improve the simulation of deformations.

Data pre-processing and post-processing

Pre- and post-processing tools allow the user to improve the appearance of a dataset. This can be done by either editing the intensity of certain pixels, voxels or larger areas of the dataset, or by changing the dataset's size or resolution.

Dataset editing

Because some features may not be distinctly separated from their surroundings, a user may want to mark or enhance structures in the dataset to improve segmentation results. The user may also want to delete unwanted features completely so that they can not interfere with the desired anatomy. Even after segmentation, the user may need to manually correct connectivity of regions or delete unnecessary parts. Editing tools perform these operations, providing the user with the ability to manually modify the dataset and geometry. For example, a segmented blood vessel may have parts missing, causing gaps in the connectivity. Therefore, allowing users to manually fill these gaps can greatly improve the results and require less effort to correct than modifying a generated mesh. Editing those few voxels can be done by selecting them in the dataset and setting a more convenient intensity. In addition to single point editing, straight lines, splines, planes, spheres, and boxes may also be provided.

Dataset manipulation

It is possible to reduce the size of a dataset by selecting a region of interest or by down-sampling the dataset. The region of interest is a tool allowing the selection of a smaller region of a dataset in three dimensions. The down-sampling method reduces the resolution of a dataset in one or more of the three dimensions. For example, by reducing a voxel size of 1x1x1 mm^3 to 2x2x2 mm^3, the dataset's size will be reduced by 8. It is important to note that such reduction will decrease the accuracy of the dataset and therefore less detail will be visible. Other manipulation methods allow the user to change the contrast of a dataset either manually or automatically. It is possible to adapt the correspondence between voxels and intensities by changing the transfer function of the dataset. This process can lead to a better contrast of the images and therefore a more efficient visualization of the dataset or a faster computation for the algorithms. This process can also be done automatically by filtering the dataset. These filters tend to smooth the images or highlight specific intensities.

Segmentation methods

Segmentation is the process of labelling voxels according to the type of material or part of anatomy [14]. Several types of segmentation methods can be recognized. These can be classified in two groups: low level approaches and model based approaches.

Low level approaches may be further subdivided into manual and threshold. Manual segmentation requires the operator to manually trace the desired structures in each slice. Thresholding techniques rely on the fact that different anatomies in a dataset typically have different intensity ranges. Therefore, applying a simple threshold to these values may remove unwanted regions of the image. Thresholding is very fast, but datasets typically contain several structures with similar values that cannot be separated. Thus, this process is more useful for removing some unwanted regions of the image before applying more complex segmentation algorithms.

Model-based approaches are either semi-automatic or completely automatic. They can give good results in less time and with less user effort than the low level approaches. However, there are tradeoffs between different segmentation methods in terms of automation, user interaction, and accuracy. Active contours or deformable models for image segmentation are increasingly popular due to their flexibility and ability to refine the segmentation result. The user specifies an initial guess for the contour, which is then moved by image driven forces to the boundaries of the desired objects. Two types of forces are considered — the internal forces, defined within the curve, are designed to keep the model smooth during the deformation process, while the external forces, which are com-

puted from the underlying image data, are defined to move the model toward an object boundary or other desired features within the image. In the parametric form, also referred to as snakes, an explicit parametric representation of the curve is used. This form is not only compact, but is robust to both image noise and boundary gaps as it constrains the extracted boundaries to be smooth. However, it can severely restrict the degree of topological adaptability of the model. In contrast, the implicit deformable models, also called implicit active contours or level sets, are designed to handle topological changes naturally. However, unlike the parametric form, they are not robust to boundary gaps and suffer from several other deficiencies as well.

An alternative segmentation algorithm, not dependent on the gradient of the dataset, is provided by hierarchical segmentation. This algorithm allows the user to interactively mark points inside and outside the desired structure until the segmentation is satisfactory. Starting from these points, the algorithm iteratively merges adjacent regions based on their intensity and therefore separates the inside and outside of the structures. Griffin's original hierarchical segmentation algorithm [16] has been adapted to extend the user's 2-D slice segmentations into an efficient 3-D segmentation [17]. Fig. 1 illustrates the process of using the hierarchical segmentation tool.

Segmentation algorithms can produce better results if they take advantage of a priori knowledge about the shapes of structures. For example, blood vessels typically have a long and narrow tubular shape. Thus, if the algorithm searches specifically for these types of shapes during segmentation, undesired structures can be ignored. A process using the Hessian matrix (a second order derivative) provides a method filtering an image based on how similar structures are to tubular vessels of a specified radius. This test is called the vesselness test. By analysing the Hessian matrix for several radii, a range of vessels can be outlined. Point distribution models (PDM) is an alternative approach that uses a priori knowledge from a set of training samples. It is based on a deformable model algorithm that attempts to find correspondences across a set of surfaces/samples. It selects one sample as

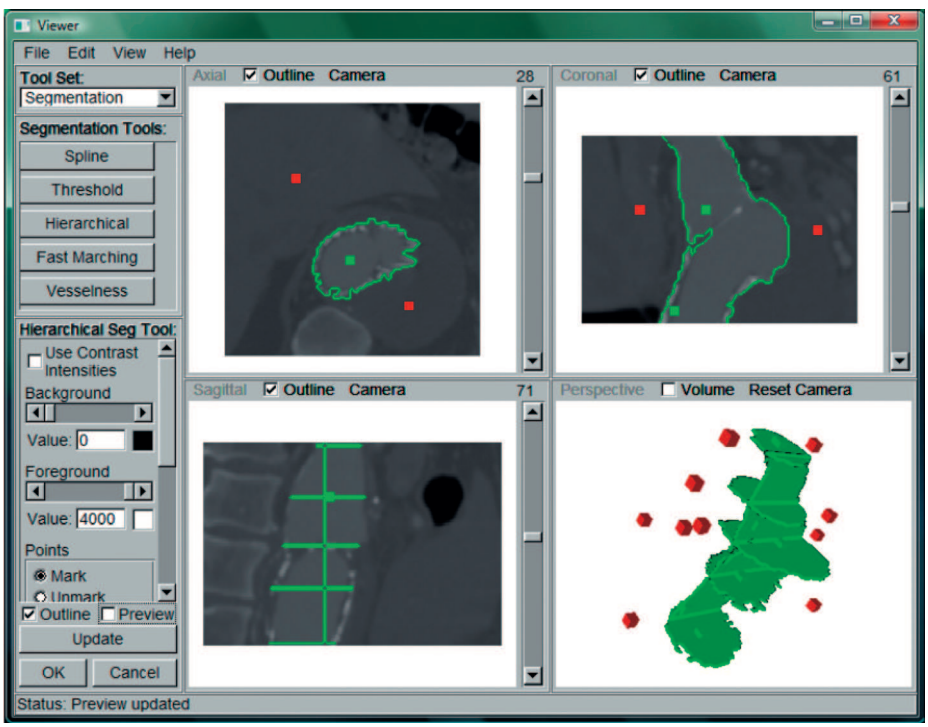

FIG. 1 Applying hierarchical segmentation by placing interior points (green) and exterior points (red). The green outline provides a preview of the generated segmentations. Notice the 2-D segmentations may be created for any slice and view.

the template, and then deforms the template to approximate all other samples. These approximations carry the correspondences from the template to all other samples. The challenge is that a single template cannot guarantee accurate approximations and therefore a statistical model or atlas is usually required.

Compact data representation

The tubular nature of blood vessels allows them to be well represented by a small set of points which follow the center of the structure. The process of finding this representative set of points and connecting them in a center-line tree is called skeletonization. The center points store the radius of the structure at their cross section, therefore they represent the volume of the vessels and the original vessels can be reconstructed from the union of each center point's sphere. The centerline structure is very useful for simulation because it allows representing the whole virtual vasculature by a small set of points and radii.

Mesh generation methods

Volumetric datasets, such as CT or MRI scans, are represented by a large number of slices and therefore require large amounts of memory for storage and long computation times for manipulation. Consequently, they are not directly suitable for real-time, interactive applications. A much more compact representation of the anatomy is required. Structures can be represented by their surfaces: triangular meshes provide a compact representation that can be efficiently rendered in real time. These meshes can be generated from the segmented datasets using the Marching Cubes algorithm or Delaunay Triangulation and then refined as needed. Some simulations also require an interior structure that fills the volume of the object; this is provided by a tetrahedral mesh that may be generated using a Delaunay tetrahedralization [18].

Surface meshing algorithms may result in a large number of elements. Since the purpose of the geometry is to provide a compact and efficient representation of the data, the number of triangles must be reduced by using simplification tools. The two commonest ones are called decimation and smoothing. Decimation reduces the number of triangles of a mesh. It controls the shape of the surface by fixing the triangles at its sharp edges while reducing the others. If the decimation chosen is too large, it might not preserve the mesh topology and result in holes or inaccurate boundaries. Smoothing improves a mesh by reducing sharp features or bumps. The most common process is the Laplacian smoothing, which performs a simple and fast relaxation of the mesh. It tries to keep the structure's features while reducing the edge angle between adjacent triangles.

Soft tissue and physiology modelling

Having built the 3-D models from medical imaging data sets as described in the previous section, it is necessary to implement the ability to interact with such models in order to be able to simulate a real life interventional procedure. Such interaction requires the detection of contact or collisions and the generation of a suitable response, both visually as well as haptically. Once a collision between an instrument and soft tissue is detected, the force being applied is calculated and fed into a mathematical model to update the new location of the tissue undergoing the interaction. In order to guarantee a realistic response, the model needs to capture the complex properties of living tissue, as well as incorporate the physiological behaviour of the organ/structures of interest. Due to the real-time requirements of simulation for training and the complexity of soft tissue, it is necessary to use strategies for optimizing the computations. Such strategies include simplification to linear models, even though living tissue is anything but linear, as well as the use of pre-computation, which limits the range and type of interactions, and, more recently, high performance GPUs. We present two main approaches for modelling soft tissue, followed by examples and a brief discussion on real-time physiology modelling.

Soft tissue modelling techniques

These can be broadly classified as physics-based and non physics based. The main difference between the two is the governing law of the model. Whereas physics-based are formulated around real physics laws, non physics-based techniques do not rely/depend on such laws, and therefore their behaviour is more free, but at the same time erratic.

Non physics-based methods

Before studying the deformation models based on real physical laws, we discuss heuristic methods

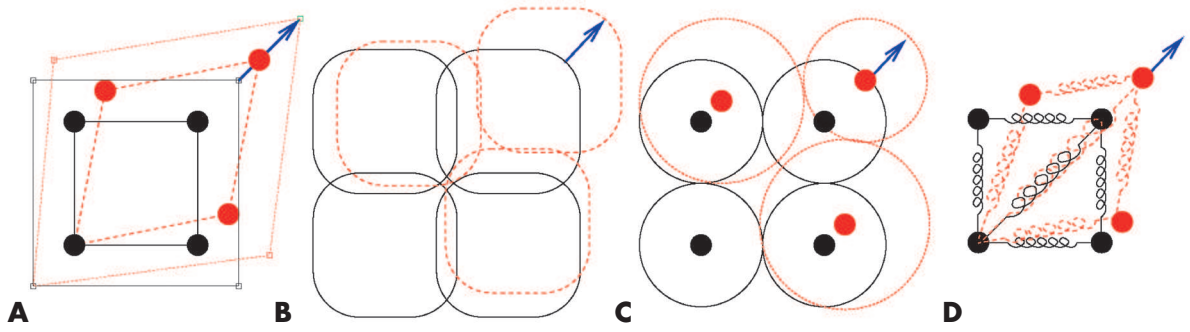

FIG. 2 Non-physics-based methods: A) Free Form Deformation; B) ChainMail; C) Particle System; D) Mass-spring.

to simulate deformations. These methods often consist of geometrical transformations and their main characteristic is their fast computation. We look at the five most popular techniques. For each method, Fig. 2 shows an illustration in 2-D of its principles on a square during the elongation of a corner. The continuous black line is the initial position and the dash red line is the final position.

The free form deformation models (Fig. 2A) consist in deforming the space where the object is defined. These deformations could be stretching, folding, torsion. Only one parallelepiped is needed to deform a 3-D object. The ChainMail method (Fig. 2B) is based on a volume discretization in cubic elements. Each element is linked as a chain. Therefore, each element can move freely without influencing its neighbour in a given limit. However, the propagation is directly performed during large deformation with the adjacent nodes. The particle system (Fig. 2C) is a set of punctual masses moving under external actions as well as internal actions resulting from intra-particle action. The particle behaviour in response to these forces is usually under physical laws. The mass-spring system (Fig. 2D) is based on a volume mesh. The nodes are punctual masses and a cohesion force is applied on each node edge in order to keep a solid unity. Commonly, this force is the linear elasticity as it is the case for springs.

PHYSICS-BASED METHODS

Physics-based modelling involves constructing dynamic models of objects and computing their motions via physical simulation. It implies that the simulation of the object's motions is governed by the laws of physics, which leads to physically realistic behaviour. This approach allows concentrating on the global behaviour of the object as a whole, since it is synthesized automatically by the physical simulation, while the previous one is more focusing on specifying low-level behaviour details (for each particle). For example, when simulating the motion of a passive object (i.e. inanimate but still subject to the external forces such as gravity), the simulator needs only the initial state of the object and the simulation engine automatically computes its motion by integrating the differential equations stemming from Newton's laws. When the object is in motion, boundary conditions are specified: location of the forces' application nodes and their components. A mathematical integration of those forces, coupled with the initial forces, leads to the definition of the motion of the object.

Because it is impossible to solve those equations for the whole object on the continuous domain, it is needed to discretize it as a mesh composed of a set of discrete sub-domains, usually called elements. This discretization allows solving the laws of physics for several key particles of the object, which leads to an approximation of the whole behaviour. Those particles are assimilated to the elements' nodes inside and outside the object to approximate the real behaviour as closely as possible. One of the most common physics based method is the finite element method [19]. It is a numerical technique for finding approximate solutions for the equations governing the laws of physics (usually under the form of partial differential equations or integral equations). The solution is computed either by eliminating the differential equation completely, or by approximating the partial differential equations with a system of ordinary differential equations, which is then solved using standard numerical techniques. The main challenge for this

approach is to approximate those equations in a numerically stable way, i.e. without accumulation of errors at each time step of the simulation. Besides the finite element method, other approaches also deal with the physics based models but they solve them differently. The fast finite element method uses a faster approach to solve the equations. The beam models represent the object by a set of beams instead of elements. And the tensor mass method discretises the object by a set of masses linked by tensors, in a similar fashion to the mass-spring models.

Examples of soft tissue modelling

Two examples of soft tissue modelling are presented below: a liver and a vascular network (Fig. 3) [20]. Both examples propose several approaches to illustrate the modelling possibilities.

The liver has been widely simulated with the techniques previously described. The deformation is interesting in a liver biopsy or within laparoscopic simulators. The mass-spring method is often used to model liver deformation because its small computation time easily allows haptic rendering. However, its heuristic formulation implies a complex tuning and liver mechanical properties determined in-vivo allows a direct parameterization with physically-based methods such as FEM. Besides, the accuracy provided by the latter would enhance the quality of the simulation, even if the haptic interaction will be harder. Finally, both methods could handle topology changes as it is required in some operation such as needle insertion, cutting, dissection, etc.

Modelling the vasculature is a key issue in most interventional radiology training simulators that require a consequent database of cases with various anatomies and pathologies to propose various training experiences and therefore improve trainees skills. The current commercial simulators are so far assuming that the vessels are non deformable. However, it is well known that the vessels deform slightly when blood is pulsating in them, and on large scale when rigid instruments are introduced in them. Several approaches can be envisaged to model those deformations.

The finite element method can be used to model the vessels and their deformations. Unfortunately, given the computational load needed, it does not work in real time in three dimensions. This method is therefore hard to be applied for training purposes where realism and real time behaviour are necessary.

The mass spring approach has also been investigated to model vessels. Zorcolo et al. [21] present a simulator where the catheter can deform the vascular surface and even go through it. It allows a realistic deformation of the surface for the given vasculature. The problem, like for most of the mass spring models, is that it is long and tedious to adjust the parameters of the springs to achieve a realistic behaviour. It is consequently difficult to extend this method to new datasets. Coupling a mass spring model and a centerline representation of the vessels could bring an efficient solution. The idea is that, instead of having particles all along the vascular surface, only a few of them on the vessels' centers are needed. When a deformation occurs,

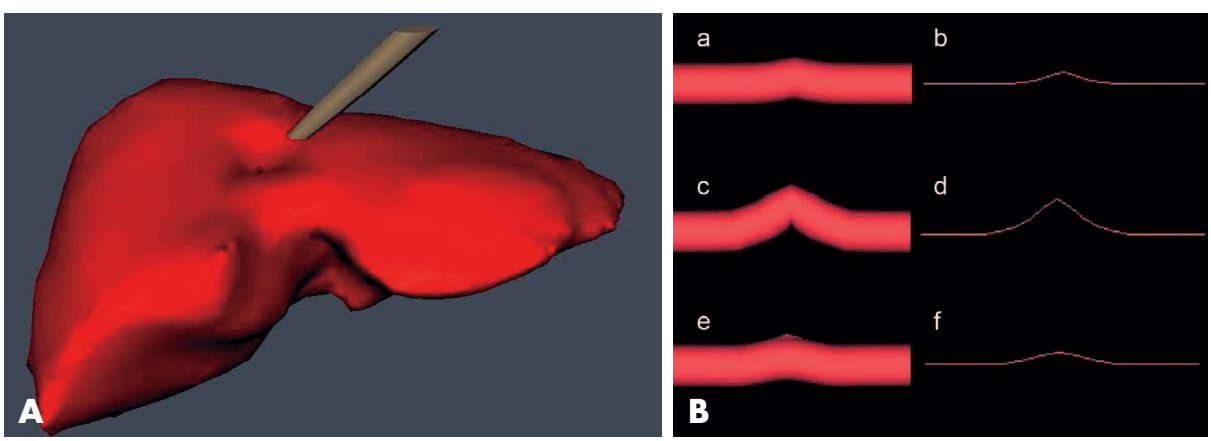

FIG. 3 A: Needle insertion in the liver with a haptic device. B) Centreline representation of a straight vessel (a, b shows small deformation, c, d shows large deformation, e, f shows small deformation with surface local deformation) [19].

the surface corresponding to a certain center points is updated while the others are not moving.

Physiology modelling

The realism of a simulator can be significantly improved by incorporating relevant physiological behaviour such as motion and flow. Adding the dynamics of respiratory or cardiac motion, and/or the flow of contrast agent, can make a major difference to the acceptability of a simulator. It is important to point out that the real time requirements of a training simulator restrict the type of physiological models that can be incorporated as a large proportion of these are non interactive and far from real time. A key aspect of physiology modelling in this context is that of clinically-based boundary conditions. These are required in order to faithfully reflect real-life clinical behaviour.

Motion

Respiration is of interest in many procedures such as radiotherapy as it may affect tumour location, or visceral interventions. Respiration is a complex process to model because it is due to the action of many muscles (abdominal, inter-costal, scalene and the diaphragm). Moreover, many organs are in interaction and for each of them it will be necessary to compute their intrinsic motion, their deformation and collision detection to avoid inter penetration.

Cardiac motion may also interfere with the behaviour of soft tissues. Every organ directly in contact with the heart moves and deforms accordingly. This is particularly the case for the aorta, the vena cava and the lungs. Integrating this motion is therefore needed to achieve a highly realistic simulator. For example, a simulator of pericardial puncture requires modelling the behaviour of the heart and its surrounding tissues to help simulate the real conditions inside the body when the needle is inserted next to the heart.

Flow

Simulating blood flow is less important than respiratory and cardiac motion in terms of interference with the tools. Blood flow, except in the aortic arch, can be neglected because of the slight motion it produces on the vascular walls. Nevertheless, in terms of visualization, a training simulator needs to model blood flow, especially when dealing with vascular interventional radiology where contrast is routinely used.

Human computer interaction

The face validity of a medical simulator greatly depends on the choices made for providing the human computer interaction (HCI). Issues to consider include whether to use an immersive environment or a mannequin based system; use of haptics devices; tracking of tools/instruments; custom build of devices; and cost. Often the actual instruments from the procedure being simulated can be taken and integrated into the training environment. As with graphics hardware, many of these HCI components have benefited from increases in performance and falling costs, and continue to do so.

Haptics

The word “haptics” derives from the Greek *haptai,* meaning the sense of touch. Haptics solutions can provide both tactile and force feedback effects, and may involve a physical or a virtual object. Use of a physical object within a simulation environment is sometimes termed ‘tactile augmentation’ or ‘passive haptics’. Here a physical model of a part of the anatomy is manufactured and the trainee can feel the object using their hands or via some tool. For example, the SKATS knee arthroscopy training system uses this approach and solid models of the femur and tibia are placed inside a knee mannequin.

In a virtual environment we need to implement an active haptics response that can change over time. A tactile response provides information on contact surface geometry, roughness, slippage, and/or temperature, and can therefore provide important cues within a medical simulation. Apart from cell phones that vibrate to provide a tactile cue to the user, the technology is not yet widespread. Pin arrays that fit onto a finger tip, and pneumatic based systems have been investigated for use in medical simulators but much work is still needed.

Hardware for force feedback is more mature and many medical simulators use either an off-the-shelf haptics device – such as the PHANToM range of robotic arms from SensAble Technologies (Woburn, USA), or a purpose built haptics device – such as the Laparoscopic Impulse Engine from Immersion Medical (Gaithersburg, USA). A good overview of force feedback haptics in medical applications can be found in reference [22]. Ideally the haptics

device will provide 6 degrees of freedom (DOF), with an active force response to both position and orientation. However, devices with 6 DOF are expensive and most off-the-shelf haptics devices only provide 3 DOF force feedback to the stylus position. In Vidal et al. [23], two 3 DOF PHANToM Omni devices are used, one of which has been set up as a virtual ultrasound transducer and the default stylus has been replaced with a mock up of a real transducer. In this example, the user interacts with the virtual patient using the haptics devices within an immersive environment. An alternative approach is to place the haptics device inside a mannequin. For example, the Medic Vision (Melbourne, Australia) Mediseus® Epidural simulator places a PHANToM Desktop robotic arm inside a mannequin and it is used to provide the appropriate resistance as the epidural needle is inserted. In many simulators, however, it is difficult to achieve the exact fidelity of the real life procedure and applying force feedback directly to the user's hand is a major challenge. In some cases however, custom-built haptic devices help improve the realism associated with a simulation. For the VEKATS simulator for example, Sherman et al. [24] used specialized instruments designed to look and feel like actual surgical tools along with a detailed computerized model of the human knee and video simulations of the surgery to provide a realistic training system for knee arthroscopy surgery.

The games market is set to provide a catalyst for further improvements in price and performance to haptics technology. For example, the NOVINT Falcon haptics device was released in 2008 costing just $ 190. The Falcon offers similar functionality to a 3 DOF PHANToM, but does have more limitations such as a smaller workspace. However, it is adequate for certain tasks in a medical simulator and is ten times cheaper than its nearest competitor.

Display technologies

If a medical simulator is based on a virtual environment then providing support for stereoscopic vision is important i.e. 3-D vision produced by the fusion of two slightly different views of a scene on each retina. There are several techniques for achieving this and stereoscopy is widely supported in medical simulators [25]. Active and passive stereo are common approaches, requiring the user to wear shutter or polarized glasses respectively. However, the requirement to wear glasses can be detrimental to the simulation. Autostereoscopy displays are therefore becoming more important as this latest generation of 3-D displays do not require the user to wear special glasses but the user must be in the correct position relative to the display, for example, such a display was used in a virtual planning environment for mastoidectomy [26].

A virtual workbench system (first proposed in 1996 by Poston et al. [27]) has been used in medical simulators to co-locate 3-D vision using active stereo with the haptics devices. Workbench systems are readily available commercially and provide a flexible platform in which to build a virtual environment. The virtual patient appears to be positioned below a semi-transparent mirror, in the same workspace as the haptic devices, which can act as a needle or other instrument. The co-location of 3-D vision and haptics greatly enhances the immersive effect. This type of environment has advantages over the use of a mannequin based simulator, such as the ability to use different data sets easily, and to provide an animated virtual patient with respiration and other physiology processes modelled, rather than just providing a virtual cadaver.

Other hardware components

In a virtual environment, position and movement of the user and tools need to be tracked in order to update the virtual images in real time. Tracking hardware is available based on magnetic, optical, ultrasonic, mechanical and inertial approaches. The choice of tracker technology depends on the task being simulated and the corresponding accuracy, speed, latency, portability and workspace requirements.

Mechanical trackers (e.g. FSL Boom) use a direct mechanical connection between the reference and the target to measure position and orientation and hence have a constrained workspace and are generally low-cost. These typically have high update rates and a good accuracy. The da Vinci simulator for interactive catheter insertion uses a hand-activated electro-mechanical catheter device for tracking.

Magnetic trackers (e.g. Polhemus *Trak series, Ascension *Bird series) are widely employed in medical simulators (e.g. VEKATS) as magnetic signals are not attenuated by the human body and hence minimize line-of-sight issues. These are easy to use and have a reasonable accuracy and latency but are susceptible to interference from electronic devices and metallic structures in the vicinity. Optical and ultrasonic trackers, while immune to these

interferences, require a clear line-of-sight between the transmitter and receiver as they rely on detection of target markers and acoustic signals respectively to measure position. Immersion Medical's LapVR laparoscopic surgery simulator uses high-speed optical tracking technology to accurately monitor tool motion. Optical trackers (e.g. Vicon 3D, Motion Analysis, etc) provide high accuracies and a low latency and can be used over a large area whereas the ultrasonic ones (e.g. Logitech 3D Mouse) typically have poor accuracy and limited range. The ultrasonic trackers are affected by noise and echoes in the environment.

Inertial trackers (e.g. Intersense InertiaCube) use gyroscope-based 3 DOF sensors to measure rotations along axes and are suited to tracking in a large workspace as they do not require a separate transmitter and are also not affected by signals or metallic objects. These are, however, not accurate for slow position changes and tend to drift over time which can be a major drawback when considering these for medical procedures. Hybrid trackers combine inertial trackers with optical (e.g. Ascension Hy-BIRD) or ultrasonic (e.g. InterSense IS-900) technology to provide drift correction, improve accuracy and lower latency.

Augmented reality interface

Augmented reality (AR) allows 3-D graphics techniques to be used to augment the reality as we see it with digital content (see Freudenthal et al. [28] for an introduction to this topic). For example, the view through a surgical microscope can be augmented with information about the location of delicate structures that are normally out of sight below the visible surface. Eventually, procedures such as ultrasound guided needle puncture could be replaced by an AR system where a 3-D rendering of the target anatomy is overlaid onto the actual patient. It will be as if the surgeon has X-Ray glasses!

Integration

Software development: from the TA to the final product

The development of a computer application is generally driven by the well-known 'waterfall model' in which the software development is considered as flowing downward like a waterfall. The development process goes through different successive steps: 1) requirement analysis; 2) design; 3) implementation; 4) integration; 5) testing; and 6) maintenance, if needed go back to step 1. The use of the CTA and validation studies are naturally suited in this development paradigm; particularly during steps 1), 2) and 5) of the waterfall model.

Indeed, the CTA for the procedure of interest is carried out by psychologist partners and it has to be extensively used to guide the design of the simulator because the CTA aims to identify, describe and detail the whole process of the medical procedure (from the cues perceived, to the cognitive thought processes of decision-making, to the ensuing physical actions). In particular, this helps to identify the minimum requirements needed within the simulator, as well as the key steps that need to be assessed as part of the training evaluation of the medical students (also called metrics). Also, the validation of the simulator is critical to its eventual uptake in a training curriculum and such projects cannot be undertaken without a close multi-disciplinary cooperation involving the domain experts, computer scientists, engineers and psychologists. The validation step also helps to further ensure that the fidelity of the simulator is at an acceptable level, and clinical validation of the system's content has to be carried out at each development stage. This can be achieved objectively using questionnaires developed to evaluate the features and the performances of the simulator. Often, the feedback given by these questionnaires leads to many suggestions for improvements of the simulator.

Simulator pipeline

The 'integration' step aims to merge both the different software and hardware components of the simulator. Indeed, different pieces of code will interact amongst each other, and also software and hardware will interact together. Care must then be given to avoid any conflicts between these different components, and the source code must be written in a standardized way (see below).

Ideally, the use of anonymized (or pseudo-anonymized), real, patient specific imaging data is recommended to allow the training simulation to take place on different anatomical regions, using accurate and variable anatomical representation, and with a range of pathological entities. This variability is important to learning. Different patient datasets are then selected and pseudo-anonymized to provide the input data of the simulator. Using automatic, semi-automatic, or manual segmenta-

tion, anatomic structures are identified. When applicable, automatic techniques are always preferred. Tissue properties are then assigned to each structure. A patient is made of several anatomic structures that are interacting. For example, ribs are moving and soft-tissues are deformed due to the respiration. This kind of interactions is performed within the 'dynamic modelling' component. During the respiration, the ribs move and the tissues are deformed: inter-penetration of anatomic structures has to be avoided. This is catered by the 'collision detection' component and the 'collision response'. When a collision occurs, the collision detection component triggers the corresponding collision response. It also takes care of detecting touch between the haptic devices (needles, catheters, guidewires, ultrasound probe, etc.) and tissues (skin or internal tissues). When a collision is detected between the patient and the haptic device, a reaction force is computed and sent back to the device. Also, the force is used to compute the deformation of the tissues due to a collision. A guidance image (fluoroscopy, ultrasound or CT) is computed depending on the position of the haptic device(s) and the state of the patient (including respiration and tissue deformations). Finally, a 'task assessment' component is included to control the user's actions according to the CTA (e.g. hit no-go area, reach the target successfully, duration of the procedure, number of needle removals/retractions before hitting the target, etc.). We saw previously that numerous components (either software or hardware) can interact between each other. The 'simulation controller' will ensure the synchronisation of these different components.

Software design

Designing a software is a key challenge for the integration team. A project is often composed of different groups with different skills to harmonize. Each entity would provide with a development code at a strategic position inside the integration framework. How could the outputs from the different centres be gathered together?

The integration of pieces of code from different research centres requires certain uniformity inside the code: it should be easily readable and understandable by all the member of the integration team. It allows the members of the project to adopt the same set of conventions. Such a point could be achieved with a common coding style.

The integration of different pieces of code in the same development project requires a tool to allow each part to be consistent together. Such tools are called 'version control system' and allow each member of the integration team to interfere with the project code at different location and at the same time. Each version of the code is then known by incrementing the revision number. Such tools are highly recognised in multi-developer project. The more famous tools are SubVersioN (SVN) and CVS.

A 'collaboration exchange website' (such as a 'wiki') is a tool to facilitate the collaborative exchange of written documents with minimum constraints. It is important in the integration process because it insures a good collaboration while every member is aware of the different current stages of each part of the project pipeline. Another useful tool is the file transfer protocol (ftp) server. Its interest is to store data too big to be on the version control system server (e.g. patient data, processed patient data). Each developer interacting in the pipeline should use the same data and if they have processed data they should put it on the ftp server for the next team on the pipeline.

The integration team should be able to understand the code or at least the input, the output and the aim of each function. This could be achieved with decent code documentation. A good tool for this is Doxygen.

Minimize incompatibilities

As it has been presented previously in this chapter, many components can be found in a simulator. It ranges from a software library for medical image segmentation to a specific piece of hardware to improve the sense of immersion of the user and to make the simulator closer to the real intervention. Setting strict design rules will clearly help integration of the software pieces together. On the same note, regular interactions between the teams developing the different modules, presented in the previous subsections, will facilitate their incorporation in the simulator. But the developers have to be careful of the possible incompatibilities between the different software libraries or programs and the different hardware devices. Three possible kinds of incompatibilities can be considered: software consideration, hardware consideration, and hardware-software consideration.

The most common source of incompatibilities is related to the software consideration. It is indeed very often the case that it is not possible to use a

library with a specific software, and this for several reasons. The obvious reason is that a library or pieces of codes communicating together are not written with the same programming language. Even though some translation functions might exist, it will lead to a considerable adaptation of the code. Another common reason for incompatibilities comes from the fact that the developers might not use the same version of a specific library. This happens often with well maintained and evolving libraries, for example Visual Tool Kit, Insight Tool Kit. It is consequently needed to decide which version to use before and to change it only if the other developers agree on it. For commercial libraries, being aware of the licence rights and the possibility of sharing them is important. Incompatibilities can also arise from the fact that the operating system running the simulator does not recognize some of the command lines within the library because it has been written and/or compiled under another operating system. For example, when a medical image processing library is compiled under Linux but then used under MS Windows, it is possible that there will be several lines of code to be rewritten, if not entire functions. Even on the same operating system, it is possible that a piece of code programmed and compiled within a certain environment might not be recognized by another environment, even though the same language is used. This can be the case between the MS Visual Studio environment and the one proposed by the GNU Compiler Collection. In all those cases, it is important to set the operating system, the environment, and the language that the simulator will be developed with in the early stages of the development. If a licence is needed for the environment or some libraries, it should be decided before the development starts, in case funding needs to be sought.

Hardware considerations can also be the source of several incompatibilities. The most common one is that the operating system running the simulator does not support the hardware chosen for the simulation. It can range from an optical mouse not being recognized by Linux to a specifically designed haptic device's driver that is incompatible with MS Windows. Most of the time, the company building the hardware is to be contacted to make sure such incompatibilities will not arise. Even if a haptic device can run on a specific operating system, its capabilities (in field of action, in force, etc.) must be checked beforehand to make sure they are able to perform the expected tasks. For example, a haptic device usually has a limited range of action (less than 10 cm usually in many haptic devices) due to its wires or due to the motors giving the force feedback. If this range of action is supposed to be large, it is reasonable to check with the hardware company if the device can handle the simulator's expectations.

Finally, like in most systems coupling two pieces of engineering, incompatibilities can appear from the link between the hardware and the software. It is indeed often the case that a hardware is sold with an interface using a specific programming language. In consequence, the main code of the simulator has either to be in the same language, or it has to have a library translating the hardware's outputs to be able to communicate. Some libraries such as H3-D propose a convenient interface between the most common haptic devices and most programming environments. It could therefore be used to bypass this compatibility problem. Nevertheless, if a specifically designed haptic device is to be used by the simulator, it is likely that H3-D would not provide any interface. A specific library should then be developed. The company developing the hardware rarely provides a versatile interface being able to be coupled with any programming language. Thus, in many cases, a specific library will need to be written to act as the interface between the hardware and the software.

In summary, to minimize the three possible kinds of incompatibilities presented above, it is worth studying thoroughly the different pieces of software and hardware to be integrated in a simulator before starting the project. Clearly stating the environment and language to be used will reduce the software problems. On the hardware side, a strict study of the capabilities and restrictions of the possible haptic devices is needed before selecting one. Then, the links between software and hardware must be clarified in order to define the way to reduce the surprises when comes the time of integrating them together to build the first simulation prototype.

Validation

Validation is usually achieved through the calibration of the model to actual system behaviour using the discrepancies between the two, and the insights gained, to improve the model. For medical simula-

tion this process is informed by subject matter expert review of content and appearances of the simulation, feeding discrepancies back to developers until model accuracy is judged to be acceptable.

Definition

To be effective, a model or simulation should be an accurate representation of the real world system that would be used for a given training objective. Hence validation starts with building the right simulation model for the task to be trained, with guidance from a detailed TA at the development stage. Once the steps of a simulation are correctly replicated (content validity), and the simulation and its context appears realistic to an operator (face validity), there is a need to go on to prove that skills learnt in the simulation will transfer to effective procedural skills in patients (transfer of training), and will be retained over time. For purposes of assessment, the simulation's assessment tools should correctly measure procedure steps that are important to safe completion of the training objective (construct validity). For predictive validity, the simulation's assessment should predict future competence in patients as confirmed in a subsequent clinical study. Lastly, concurrent validity correlates the new method with a gold standard such that the performance of experts can be distinguished from that of novices.

Examples

Validation of the laparoscopic trainer, minimally invasive surgical trainer (MIST) virtual reality has been successful in showing skills transfer to patient procedures [29], with similar success in the fields of colonoscopy, anesthetics and sinus surgery. These successes in validation have, however, yet to be reproduced for IR endovascular simulations [30] of which there are now a small number of commercial producers.

Limitations and need for relevance

Assessments provided in current endovascular simulators generally use surrogate measures of performance such as the time taken to 'successfully' perform a procedure, fluoroscopy time, 'C' arm handling or contrast volume used. While providing some indication of proficiency, these 'birds eye view' metrics give little information on detailed skills or errors. Metrics that reflect objectively determined critical procedure steps, including fine motor actions, will be more likely to discriminate between experts and untrained novices and should in time address these limitations. Simulations with the potential for higher fidelity where this is needed, and using evidence-based real world content, may well improve the likelihood, and range, of skills transference to patients.

Importance of human factors in development

In this chapter we have outlined the development of simulations to train imaging guided needle puncture using a bottom up approach based on the findings of a hierarchical, CTA of real procedures to inform simulator design. Subject experts identified metrics from this TA with a significant correlation ($p < 0.001$) between raters, indicating expert agreement on key points of the procedures. The development phase of the simulator has been supported by multiple feedback sessions with clinicians (Fig. 4), both within the academic environment, and using questionnaires to explore face and content validity at large conferences. Consulting with such a wide range of end-users has driven revisions to improve functionality, and is regarded

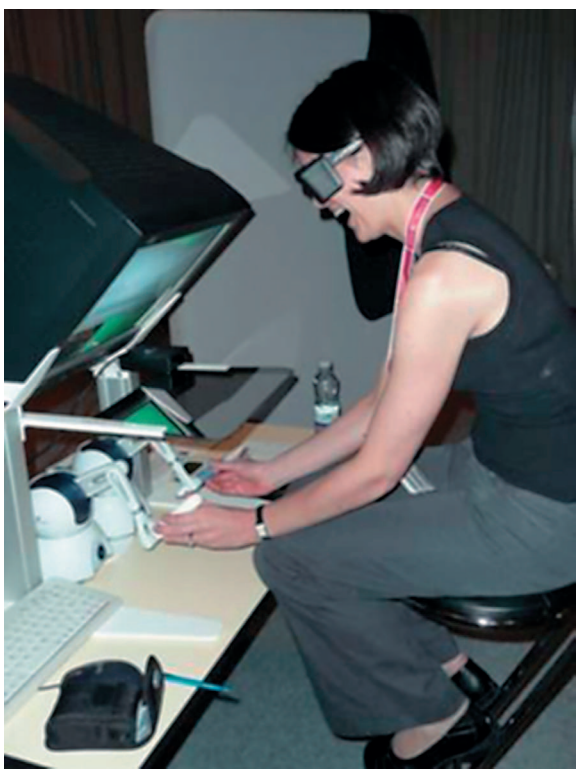

FIG. 4 ImaGiNe-S: Imaging Guided Needle access Simulation during validation at CIRSE 2008.

as essential to successful simulator development. Repeated content validation will continue until the simulator is regarded as ready for construct validation to objectively determine whether the correct procedural elements are indeed being trained and assessed. This will then be followed by transfer of training studies.

CURRENT AND FUTURE STATUS

Much of the impetus to develop medical simulation has been through concerns regarding patient safety and indeed a range of specialties is now using simulation to train procedural skills. While it may seem attractive to use simulation to train complete, complex procedures such as carotid stenting, the technology to train the finer motor skills which are central to many real-world critical performance steps, remains challenging. Limitations include compromises over computational load and the lack of data on deformations of instruments and living tissues. However, with knowledge of their limitations, and given a predetermined level of the trainee's prerequisite knowledge and skills, simulations can be used to train those cognitive and motor skills that they are known to correctly replicate. After attaining a predetermined level of proficiency in these areas, the trainee can commence mentored learning in patients.

Until simulator technology is able to correctly replicate all of a training objective's component behaviours, mentored training in patients will remain necessary to acquire the full set of skills for safe practice of procedural medicine. There is now a need for training organisations to take a lead in setting standards, including validation standards, for the use of medical simulation as an integral component of procedural curricula.

REFERENCES

1 Dodd A, Riding M, John NW. Building realistic virtual environments using Java and VRML. In: *Proceedings of Third Irish Workshop on Computer Graphics*. Dublin. 2002. pp. 53-61.

2 Anonymous. National Patient Safety Agency, Rapid Response Report: risks of chest drain insertion - *Reference NPSA/2008/RRR003*.

3 Ackerman MJ. The Visible Human Project. *Proceedings of the IEEE* 1998; 86, 3: 504-511.

4 Liu A, Tendick F, Cleary K et al. A survey of surgical simulation: applications, technology, and education. *Teleoperators & Virtual Environments* 2003; 12: 599-614.

5 Okamura AM, Simone C, O'Leary MD. Force modeling for needle insertion into soft tissue. *IEEE Trans Biomed Eng* 2004; 51: 1707-1716.

6 Azar T, Hayward V. Estimation of the fracture toughness of soft tissue from needle insertion. *Lecture Notes in Computer Science* 2008; 5104: 166-175.

7 Neequaye SK, Aggarwal R, Van Herzeele I et al. Endovascular skills training and assessment. *J Vasc Surg* 2007; 46: 1055-1064.

8 Van Herzeele I, Aggarwal R, Choong A et al. Virtual reality simulation objectively differentiates level of carotid stent experience in experienced interventionalists. *J Vasc Surg* 2007; 46: 855-863.

9 Militello LG, Hutton RJ. Applied cognitive task analysis (ACTA): a practitioner's toolkit for understanding cognitive task demands. *Ergonomics* 1998; 41: 1618-1641.

10 Grunwald T, Clark D, Fisher SS et al. Using cognitive task analysis to facilitate collaboration in development of simulator to accelerate surgical training. *Stud Health Technol Inform* 2004; 98: 114-120.

11 Velmahos GC, Toutouzas KG, Sillin LF et al. Cognitive task analysis for teaching technical skills in an inanimate surgical skills laboratory. *Am J Surg* 2004; 187: 114-119.

12 Schraagen JM, Chipman SF, Shalin VL. Introduction to Cognitive task analysis. In: Schraagen JM, Chipman SF, Shalin VL, editors. *Cognitive Task Analysis*. Mahwah, NJ. Lawrence Erlbaum Associates; 2000. pp. 3-23.

13 Clark RE, Estes F. Cognitive task analysis for training. *Int J Educ Res* 1996; 25: 403-417.

14 Vidal FP, Bello F, Brodlie KW et al. Principles and applications of computer graphics in medicine. *Computer Graphics Forum*, vol. 25, no. 1, pp. 113-137, March 2006.

15 John NW, McCloy RF. Navigating and visualizing three-dimensional data sets. *The British Journal of Radiology*, vol. 77, no. 918, pp. S108-S113, June 2004.

16 Griffin LD, Colchester ACF, Roll SA et al. Hierarchical segmentation satisfying constraints. *Proceedings of the British Machine Vision Conference*, vol. 5, no. 1, pp. 135-144, September 1994.

17 Luboz V, Song Y, Din N et al. Segmentations of 3D vasculatures for interventional radiology simulation. *Proceedings of the Workshop of British Machine Vision Association (BMVA)*. London, UK. December 2008.

18 Si H, Gartner K. Meshing piecewise linear complexes by constrained Delaunay Tetrahedralizations. In: *Proceedings of the 14th International Meshing Roundtable*. 2005; 14: 147-164.

19 Zienkiewicz OC, Taylor RL, Zhu JZ. The finite element method, its basis and fundamentals. Oxford. Elsevier; 2005. pp. 733.

20 Chen F, Gu L, Huang P et al. Soft tissue modeling using nonlinear mass spring and simplified medial representation. In: *Proceedings of the 29th Annual International Conference of the IEEE EMBS*. 2007; 5083-5086.

21 Zorcolo A, Gobbetti E, Zanetti G et al. Volumetric virtual environment for catheter insertion simulation. Eurographics Association, Springer-Verlag, Center for Advanced Studies, Research and Development (CRS4) *http://www.crs4.it*, 2000.

22 Basdogan C, De S, Kim J et al. Haptics in minimally invasive surgical simulation and training. *IEEE Comput Graph Appl* 2004; 24: 56-64.

23 Vidal FP, John NW, Healey AE et al. Simulation of ultrasound guided needle puncture using patient specific data with 3D tex-

tures and volume haptics. *Computer Animation and Virtual Worlds* 2008; 19(2): 111-127.

24 Sherman KP, Ward JW, Sherman VY et al. Surgical trainee assessment using a VE knee arthroscopy training system (VE-KATS): Experimental results. *Proceedings of Medicine Meets Virtual Reality.* IOS Press, 2001. pp. 465-470.

25 John NW. Using stereoscopy for medical VR. In: *Medicine Meets Virtual Reality 02/10, Studies in Health Technology and Informatics.* IOS Press 2002, ISSN: 0926-9630, ISBN 1-58603-203-8. pp 214-220.

26 Jackson A, John NW, Thacker NA et al. Developing a virtual reality environment in petrous bone surgery. *Otol Neurotol* 2002; 23: 111-121.

27 Poston T, Serra L. Dextrous virtual work. *Communication of the ACM* 1996; 39: 37-45.

28 Freudenthal A, Samset E, Gersak B et al. Augmented reality in surgery ARIS*ER, research training network for minimally invasive therapy technologies. *Endoscopic Review* 2005; 10: 5-10.

29 Seymour NE, Gallagher AG, Roman SA et al. Virtual reality training improves operating room performance: results of a randomised, double-blinded study. *Ann Surg* 2002; 236: 458-464.

30 Gould DA, Kessel DO, Healey AE et al. Simulators in catheter based interventional radiology: training or computer games? *Clin Radiol* 2006; 61: 556-561.