

A prototype percutaneous transhepatic cholangiography training simulator with real-time breathing motion

Pierre-Frédéric Villard, Franck P. Vidal, Carrie Hunt, Fernando Bello, Nigel W. John, Sheena Johnson, Derek A. Gould

► **To cite this version:**

Pierre-Frédéric Villard, Franck P. Vidal, Carrie Hunt, Fernando Bello, Nigel W. John, et al.. A prototype percutaneous transhepatic cholangiography training simulator with real-time breathing motion. International Journal of Computer Assisted Radiology and Surgery, Springer Verlag, 2009, 4 (6), pp.571-578. <10.1007/s11548-009-0367-1>. <hal-00430223>

HAL Id: hal-00430223

<https://hal.archives-ouvertes.fr/hal-00430223>

Submitted on 6 Nov 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Prototype Percutaneous Transhepatic Cholangiography Training Simulator with Real-time Breathing Motion

P.F. Villard^a, F.P. Vidal^b, C. Hunt^c, F. Bello^a, N.W. John^b, S. Johnson^c, D.A. Gould^d

^a*Imperial College, London, UK*

^b*Bangor University, UK*

^c*Manchester Business School, UK*

^d*Royal Liverpool University Hospital, UK*

Abstract.

Purpose: We present here a simulator for interventional radiology focusing on percutaneous transhepatic cholangiography (PTC). This procedure consists of inserting a needle into the biliary tree using fluoroscopy for guidance.

Methods: The requirements of the simulator have been driven by a task analysis. The three main components have been identified: the respiration, the real-time X-ray display (fluoroscopy) and the haptic rendering (sense of touch). The framework for modelling the respiratory motion is based on kinematics laws and on the Chainmail algorithm. The fluoroscopic simulation is performed on the graphic card and makes use of the Beer-Lambert law to compute the x-ray attenuation. Finally, the haptic rendering is integrated to the virtual environment and takes into account the soft-tissue reaction force feedback and maintenance of the initial direction of the needle during the insertion.

Results: Five training scenarios have been created using patient specific data. Each of these provides the user with: variable breathing behaviour, fluoroscopic display tuneable to any device parameters and needle force feedback.

Conclusions: A detailed task analysis has been used to design and build the PTC simulator described in this paper. The simulator includes real-time respiratory motion with two independent parameters (rib kinematics and diaphragm action), on-line fluoroscopy implemented on the Graphics Processing Unit (GPU) and haptic feedback to feel the soft-tissue behaviour of the organs during the needle insertion.

Keywords: interventional radiology; virtual environments; respiration simulation; x-ray simulation; needle puncture; haptics; task analysis

1. Introduction

Interventional radiology is minimal access intervention where medical imaging is used to guide the manipulation of instruments within various organ anatomies to perform a range of therapeutic procedures. Such interventions generally have lower risk for the patient than the equivalent surgical procedure, with less post-operative pain and shorter recovery times. The techniques used involve guidance of needles into viscera to target duct and vessel anatomy, including the gall bladder, bile ducts or portal vein in the liver, the urinary collecting system, the gut and blood vessels. One of these procedures, percutaneous transhepatic cholangiography (PTC), uses needle accesses into bile ducts within the liver tissue for diagnostic and therapeutic purposes. For diagnosis, the procedure uses X-ray imaging (fluoroscopy) during injection of radiopaque contrast medium to identify the presence of obstruction of bile due to a stone or tumour. With current noninvasive techniques in biliary imaging (computed tomography, CT; magnetic resonance, MR) such invasive techniques are rarely, if ever, needed for diagnostic purposes. For therapeutic interventions in biliary obstruction, however, placement of a stent into the biliary tree using these techniques has become a mainstay of palliative interventions. The skills required to perform such a procedure are typically acquired during an apprenticeship in patients. Inexpert manipulations of the trainee, however, can produce pain and complications, as well as increasing the procedure time. Further, the loss of PTC as a diagnostic procedure has led to a dearth of these straightforward cases for training purposes.

Simulation has been proposed as an alternative to patient based training of procedural skills. There are a number of commercial surgical simulators, mainly for laparoscopic basic and procedural skills, though there is at present no available PTC simulator. Symbionix indicates that a PTC module for their PercMENTOR is under development¹, but not yet available. In [1], a Percutaneous Renal Access Simulator is described and it incorporates breathing motion. The task in PTC differs from this one which simulates renal stone extraction, rather than liver needle insertion with contrast injection into bile ducts. In addition, current commercial simulators either do not incorporate respiratory motion or do so in a fixed, simplistic manner that does not take into account the non-reproducibility nature of the breathing cycle. Respiration can confound an operator's attempts to perform a correct needle placement, hence our simulation replicates these actions with a facility to engage 'breath-hold' at the request of the operator. Respiratory motion is due to the action of a set of muscles that are acting synchronously. These cyclic behaviours are modelled by a sinusoid in our model. However each muscle can influence breathing to different degrees of magnitude that can vary with the phase of respiration. For instance a patient could have a given tidal range due to specific thoracic and diaphragmatic muscle actions, but could maintain the same respiratory volume and frequency during a sudden change to use of the diaphragm alone. By considering just the two most important muscles in this process, we can model these effects and so test the trainee's skill to anticipate variable organ motion.

We propose a computer-based simulation to train the procedural skills to perform PTC, using optimised algorithms for simulating respiration and fluoroscopy. A corresponding task analysis has been carried out to identify the key elements that must be included into the virtual environment and also the level of fidelity required for each component of the simulator. The first aspect to model is the respiratory motion given its importance in the behaviour of visceral motion during the procedure. The second aspect is the simulation of the fluoroscopy to guide needle placement and monitor contrast injection. The last aspect is modeling the sense of touch during needle insertion. In this paper we demonstrate how to implement these aspects in our simulator using five virtual environments that are available to use for training purposes.

¹ http://www.symbionix.com/PERC_Modulles.html

2. Methods

2.1 Task analysis (TA)

2.1.1 Role of the Task Analysis

Task analytic techniques have informed much of the health and safety legislation in operation today [2]. 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.* [3] described the use of cognitive TA in the development of surgical training, and Velmahos *et al.* [4] applied it to the teaching of technical skills within surgical skills laboratories.

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, often in the form of a flow diagram. A complete TA will detail, step-by-step, each point that needs to be conducted in order to successfully complete a task. TA originally focused exclusively on observable movements and completion of steps required to achieve a goal or complete a task. However, where tasks are complex and decision making is a core component of a task, a more detailed mapping of operators' thought processes was required to fully understand and describe the completion of tasks. Hierarchical TA, an extension of traditional TA, and then Cognitive TA were therefore developed [5].

Data was collected in this study through: (1) discussions with CRaIVE² 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, and (4) interviews with subject matter experts.

2.1.2 Importance of Respiration

Respiration is a key factor in the development of Interventional Radiology (IR) simulators in order to ensure realism of design. Experts who were interviewed to inform the Cognitive Task Analysis referred to respiration at key points of the PTC procedure. For example, when injecting local anesthetic, the operator can determine the position of the needle by noting respiratory movement. If the needle is moving back and forth with a patient's respiration this indicates that the needle tip is in the liver or other organ (e.g. kidney) that moves with respiration. This movement is therefore an indication regarding the position of the needle.

2.1.3 Importance of fluoroscopic display

The fluoroscopic display can be checked at various points within the PTC procedure. For example, firstly when identifying the puncture site. The operator places a long pair of forceps on the patient's side where the liver is expected to be located, i.e. halfway between front and back, and between two ribs. The operator will check the screen and ensure the forceps tip location is below the level of the diaphragm and adjacent to the liver. Checking the fluoroscopic display throughout the procedure will inform the operator of the position and the advancement of the needle, therefore it is an essential part of the PTC procedure.

² Collaborators in Radiological Interventional Virtual Environments (www.craive.org.uk/)

2.1.4 Importance of haptic feedback

Having the ability to sense the feeling of touch in a virtual environment will inevitably increase the realism of simulator design. Haptic feedback also provides important cues for the user. The importance of haptic cues is particularly noted when advancing the 21G needle in a PTC procedure. For example, after penetration of the skin and subcutaneous tissue, the resistance to the needle increases as it travels through the body wall. When the operator feels the resistance drop, the needle has exited the body wall. The operator must immediately alter the needle trajectory moving it slightly cranially (upwards towards the patients head). The operator will continue to advance the needle into the liver and will usually feel a ‘pop’ as the needle enters the liver. All IR procedures are reliant on the sense of touch; therefore haptic feedback is an essential feature in IR simulation.

2.2 Breathing simulation

Before presenting our modelling method, let us first focus on the anatomy and the physiology [6]. Breathing consists of inflating and deflating the lungs depending on the respiration muscles. The main muscles involved are the intercostals and the diaphragm. The former allow the ribcage to expand or retract, whilst the latter contracts to draw down and inflate the lung bases or, conversely, relaxes.

The breathing process can be simplified to be cyclical and can be modelled using a sinusoid. The respiratory rate can vary during the procedure and such variability is supported in our simulator, from a total breath hold, to hyperventilation, through an intermediary stage of tidal breathing. The intercostals and the diaphragm have an independent action which is reproduced in our simulator. A patient may breathe using the diaphragm alone, or may use 75% intercostal contraction and 25% diaphragm action. For this reason, two time-dependant functions control the two anatomical muscle groups.

Based on the work presented in [7], we divide the ribcage into two parts: i) the spine modelled as a rigid static object, and ii) the ribs modelled as rigid objects that follow a kinematic law. Each rib motion is composed of two rotations defined by two angles: γ represents the lateral excursion variation of the ribs and λ the anteroposterior diameter variation of the thorax (see Fig. 1). They are defined by their maximum value and we assumed that their value is linear with the time such as:

$$\lambda(t) = \lambda_{\max} * 1/2 * (1 + \sin(2\pi ft)) \quad (1)$$

$$\gamma(t) = \gamma_{\max} * 1/2 * (1 + \sin(2\pi ft)) \quad (2)$$

with f the respiration frequency and t the time.

The diaphragm is mainly composed of muscle fibres that contract during inhalation, with a central tendon that is assumed to be rigid. The tendon motion has an upward and downward movement (see Fig. 1). It is controlled by the force F_d as follows:

$$F_d(t) = F_{\max} * \sin(2\pi ft) \quad (3)$$

The diaphragm soft tissues follow the anatomy elements whose motions are known, i.e. the rib rotation and the central tendon translation.

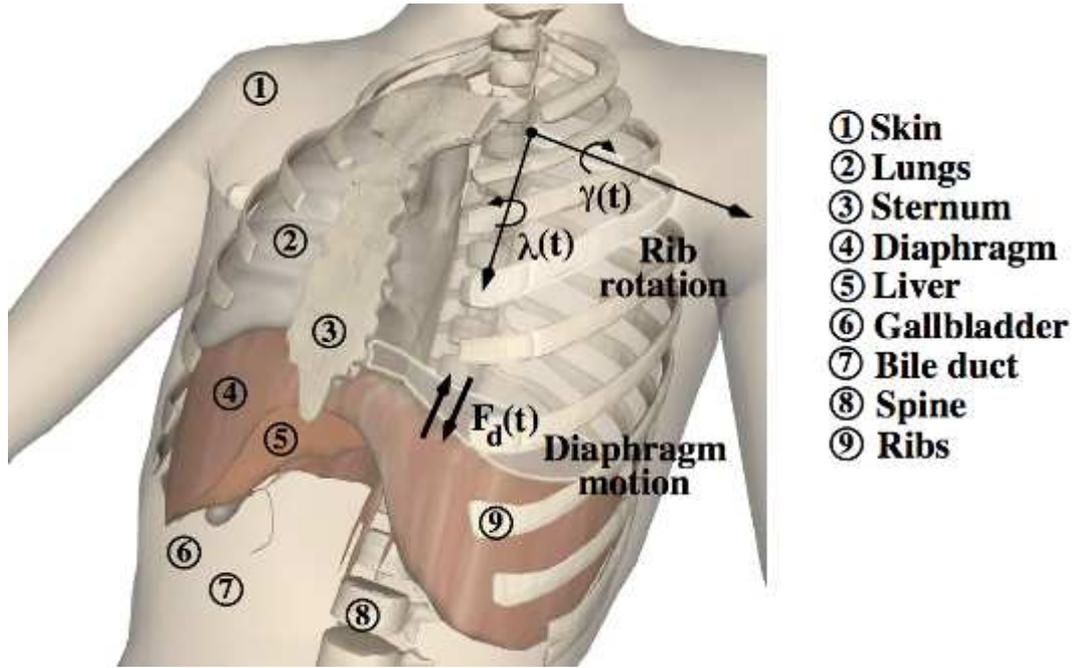


Fig. 1. Respiration modelling: rib kinematics and diaphragm forces parameterisation

We extended the 3D ChainMail algorithm [8] to model soft tissue behaviour. The original algorithm consists of dividing the tissues into small elements that are interconnected as links in a chain. Each element can move freely without influencing its neighbours within a certain limit. When an element reaches this limit, the neighbours are also moved in a chain reaction that is governed by the stiffness of the links. Mechanical parameters such as compression, stretching and shearing can be defined to control the ChainMail behaviour.

Many organs, particularly in the abdomen, move during respiration. Starting with the muscle actions previously analysed, we present how the connection is achieved for other organs. Adjacent organs share common areas that are deformed similarly. To speed up the deformation process, we make use of this property and link these areas together by vertices defined by the closest region and they are moved simultaneously. This approach requires to first identify candidate linked regions whose distance must be below a given threshold.

2.3 Fluoroscopy Simulation

2.3.1 Principles

Fluoroscopy makes use of x-rays to produce real-time video images. When photons cross matter, they can interact with matter according to various mechanisms. However, only directly transmitted photons are essential in fluoroscopy because they project on the detector an image of the human body. Let us consider the heterogenous object presented in Fig. 2. Direct radiation is computed using the Beer-Lambert law (also called attenuation law) as follows:

$$\int N_{out}(E)dE = \int N_{in}(E) \times \exp\left(-\int \mu(\rho(x), Z(x), E)dx\right)dE \quad (4)$$

with $N_{in}(E)$ the number of incident photons at energy E , $N_{out}(E)$ the number of transmitted photons and μ the linear attenuation coefficient (in cm^{-1}). μ depends on: i) E - the energy of incident photons, ii) ρ - the material density of the object, and iii) Z - the atomic number of the object material.

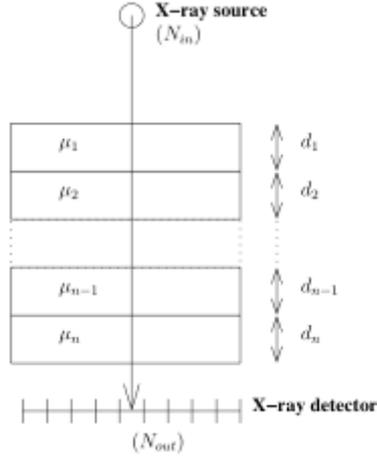


Fig. 2. X-ray attenuation through a heterogeneous object.

The discrete form of the attenuation when the incident x-ray beam is monochromatic (i.e. all the incident photons have the same energy), is:

$$N_{out}(E) = N_{in}(E) \times \exp\left(-\sum_{i=1}^n \mu_i d_i\right) \quad (5)$$

We use this simplified form to simulate visually convincing fluoroscopic images.

2.3.1 X-ray Simulation Algorithm

The simulation of directly transmitted photons can be achieved using ray-tracing. A ray is cast between the x-ray source to every pixel of the detector. Every intersection between a ray and an object is taken into account to compute the thickness penetrated by the ray into the object. Freud *et al.* demonstrated how to efficiently compute the path length of the ray crossing a polygon mesh [9]. However, the simulation is performed on the Central Processing Unit (CPU) and this prevents the simulation of complex geometry in real time. We present here how this can be tackled using the capability improvement of today's graphics card.

For GPU programming, we split Eq. (5) to enable multi-pass rendering. Each sub-equation is computed in successive rendering passes. It makes use of off-line rendering using framebuffer objects (FBOs). Intermediate rendering passes are not displayed on the screen, but stored into 2D texture maps associated with the FBOs. Fig. 3. shows the simulation pipeline. For each object, the first pass evaluates the path length ($Lp(i)$). It makes use of a sum operation that is efficiently achieved on the graphics card using the built-in blending function. For each object, the second pass makes use of the first pass and it consists of computing the product $\mu_i Lp(i)$ and simultaneously updating the result of the sum $\sum \mu_i Lp(i)$. Only the result of the sum needs to be stored. Once these operations have been performed for each object, the attenuation is computed using the result of the sum. This final rendering pass is displayed directly on the screen.

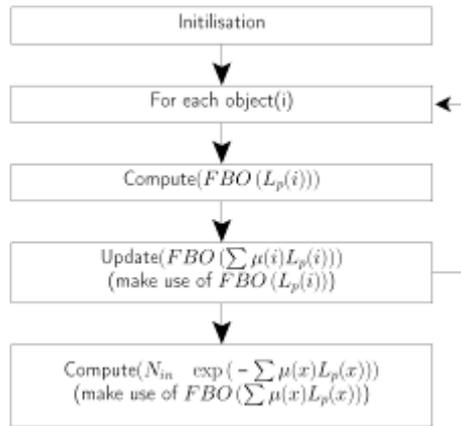


Fig. 3. X-ray simulation pipeline.

Weaknesses leading to artefacts have been identified in the algorithm to compute path lengths. When intersections occur between a ray and an object, there are the same number of incoming and outgoing intersections. Intersections can be duplicated when the ray hits triangle edges or vertices and uncertainty occurs when the normal vector to the object surface is perpendicular to the ray. Freud et al. demonstrated how to tackle these but their implementation is not easily portable on graphics hardware. However, during the path length computation, it is possible to compute on the GPU the sum of incoming and outgoing intersections. If their numbers differ, an image smoothing filter is used locally to correct the artifacts in the path length image. Fig. 4 shows real-time fluoroscopy images produced from a breathing patient.

2.4 Haptic feedback

The sense of immersion in our simulator is given by 3D stereoscopic glasses and by the use of a haptic device. The latter is used to control the insertion needle position (3 degrees of freedom) and direction (3 degrees of freedom). Moreover haptic feedback is provided to the user. The effect is that the user can feel the different tissue layers as in a real procedure.

First, the bones are rigid bodies and a stiff contact is provided when the needle touches the spine or the ribs. To improve the robustness of the collision detection while the organs are moving, we use the Ruspini algorithm [10].

Then we provide a force to constrain the needle to follow a straight line once the length of the shaft inserted reaches a given threshold. Indeed, it is generally difficult to change the direction of a needle deeply placed in a patient. However, it is possible to change the direction when the needle tip is within, or withdrawn to, a superficial level in the tissues. Practically, the initial insertion position \mathbf{P}_i and the initial direction \mathbf{D}_i of the needle are stored once the user starts to push the needle into an organ. Once the threshold is reached, \mathbf{D}_i is constant. During each haptic rendering loop a correction force is computed so the haptic avatar position remains on the line defined by \mathbf{P}_i and \mathbf{D}_i .

Finally, the soft tissue response is computed for each organ punctured by the needle. The force feedback has been implemented as in [11]. It behaves as a succession of exponential-like rises before and after liver capsule penetration. The maximum force is felt when movement ceases and it is proportional to the depth of penetration. During the relaxation phase, a slow decreasing repulsion force is applied along the axis of the needle even though it is not moving. Such force depends on the bevel of the needle.

2.5 Virtual Environments

One of the main outputs of our simulator is the possibility to change the anatomy, either to train the radiological interventionist with different anatomies, or to allow practice using a specific

patient's dataset where the patient concerned is to be the subject of an intended future procedure; this prior practice is referred to as 'mission rehearsal'.

With patient consent and ethics approvals, a large number of patient datasets have been selected at the Royal Liverpool and St Mary's Hospitals and uploaded to an on-line data repository. This protocol provides us with pseudo-anonymous CT and MRI 3D images that correspond to real cases and include pathologies of interest for training.

These data are then pre-processed to be used in our virtual environment. This consists of segmenting each organ that is important to correct content within the simulation. These segmented structures are then meshed and converted into the X3D format.

Currently, 5 patients are available (see Fig. 4). For each of these, a virtual environment has been built with the anatomic data of the patient. These data are represented in a scene graph written in the X3D format. The complete patient model contains the ribs, spine, sternum, diaphragm, lungs, cartilage, liver, the gallbladder, the bile ducts and the skin. The VE provides the user with a 3D representation of the patient, respiration and fluoroscopy, both in real-time.

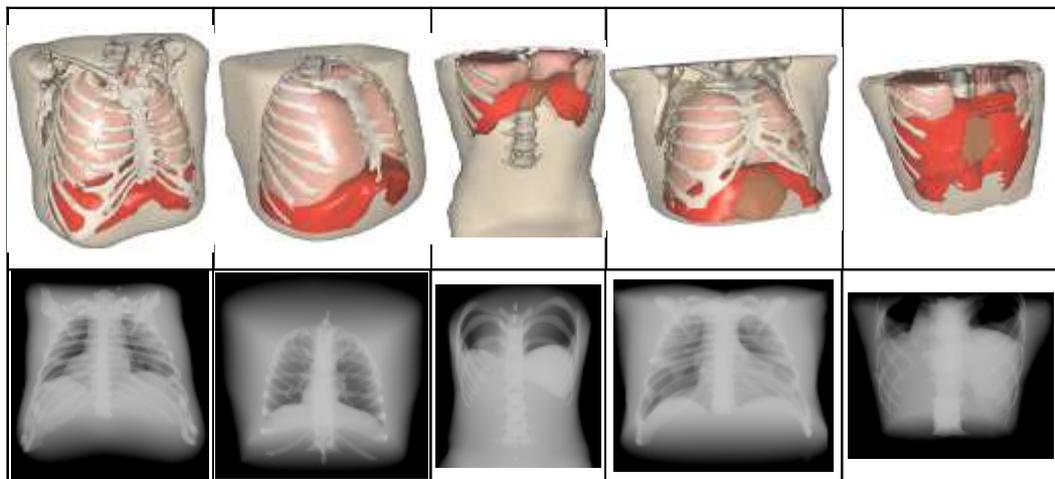


Fig. 4: Virtual environments for 5 patients. Above: mesh representation, below: fluoroscopic rendering.

3. Results

3.1 Training simulator

Our algorithms have been coded in C++ and use the H3D API. All calculations are performed using the CPU, except the fluoroscopy simulation task, which is done in parallel on the GPU. The graphic rendering is performed at 70 FPS while 1024 Hz are maintained for the haptic rendering. Moreover we are using off-the-shelf haptic devices and there is no customised haptic device available as yet.

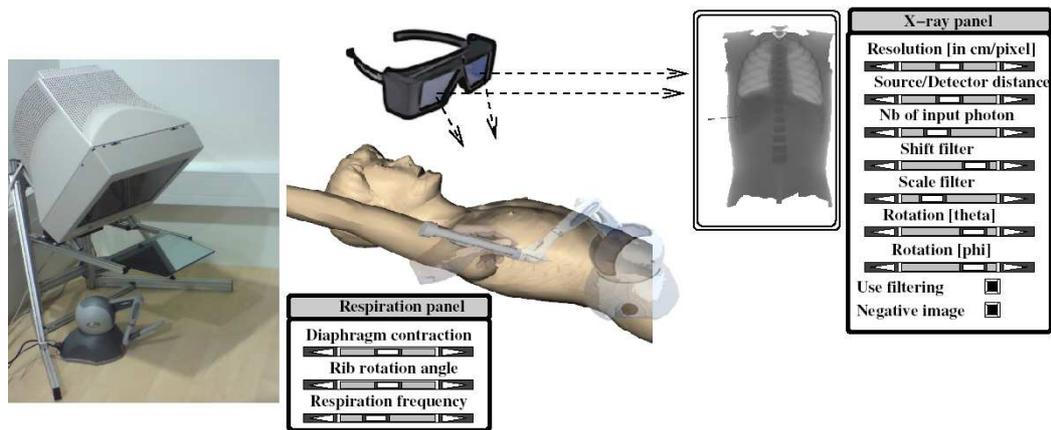


Fig. 5. PTC Simulator *left*: hardware, *right*: haptic device and 3D glasses.

Fig. 5 shows how the user can interact with the simulator. Stereoscopic 3D shutter glasses allow the user to see the patient skin in 3D and help him with the space navigation of the needle. Similarly to the operating theatre, the 2D fluoroscopic image is displayed on a dedicated monitor. With one hand the user is handling the needle via the haptic device. A force feedback is rendered to the user while inserting the needle. Finally, a control panel is available to control the respiration behaviour and the fluoroscopic rendering.

The control panel focusing on the respiration is composed of 3 sliders. The first one controls the diaphragm influence by changing the amount of the maximum displacement of the central tendon. The second one controls the rib cage influence by changing the amount of the maximum rotation angle of the ribs. The last one controls the respiration rate in order to go from a tidal breathing to a hyperventilation state. Finally, 2 hotkeys are available to emulate asking the patient either to stop breathing or to carry on breathing. The panel of the fluoroscopy simulation controls the x-ray tube and the detector positions, the energy of the incident beam and the intensity of the image (i.e. the number of photons).

Two interventional subject experts reviewed a simulation of diaphragm motion using fluoroscopy. This was considered a realistic simulation, though comments included a need for greater interpolation for smoother motion, with the more 'quiet' respiration sequence being typical of that seen during actual PTC procedures.

3.2 Respiratory motion simulation

As previously seen, visceral motion is principally induced by the diaphragm. In order to validate its motion, we performed a qualitative evaluation looking at the initial and final stages of respiratory motion. Figure 6 illustrates the simulation results for one of our datasets. Diaphragm geometry at the beginning of the simulation (end of inhale) is represented by a wireframe. The final geometry (end of exhale) is represented by the coloured surface. The physiological behaviour of this model was discussed with our clinical collaborators who validated the accuracy of the following key points: i) The muscle relaxation increases the height of the domes of the diaphragms; ii) the rib rotation compresses the diaphragm on each side and iii) the sternum compresses the diaphragm as it moves posteriorly.

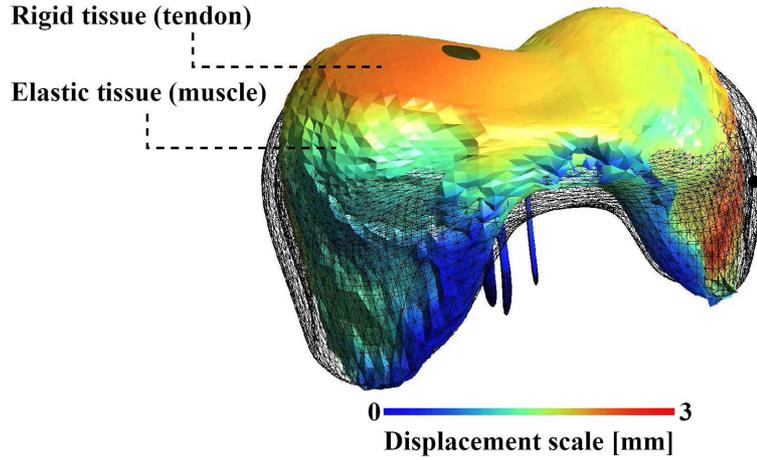


Fig. 6. Illustration of diaphragm motion during inhalation.

3.3 X-ray simulation

To validate the fluoroscopy simulation approach, we compare results produced using our GPU-based implementation with results based on a standard X-ray simulation program running on CPU based on [8]. This latter one consists in computing the photon attenuation based on physical laws and is extensively validated in the particle physics community. The purpose of the comparison is twofold: firstly, we measure the pixel disparity between images computed on GPUs using full floating point precision (128 bits per pixel) and half floating point precision (64 bits per pixel) with a reference image computed on the CPU implementation. Disparity measurements are presented in Table 1. It shows that the relative error stays below 1.2%. The numerical disparity is therefore negligible.

Table 1: Disparity measurements.

Precision	GPU	Maximum error	Average error
full float	GeForce 8800 GTX	2.55e-3	2.19e-6
half float	GeForce 8800 GTX	1.20e-2	1.36e-3
full float	GeForce 8600M GT	2.55e-3	2.22e-6
half float	GeForce 8600M GT	1.20e-2	1.36e-3
half float	Quadro FX 3500	1.12e-2	1.36e-3

Secondly, we compare the computing time. A predefined animation of 1000 frames is generated using test objects with 11102, 47794, 202520 and 871414 triangles (See Figure 7). In the case of objects with 871414 triangles, the GPU implementation is up to 65 times faster than the CPU implementation. With the least powerful GPU used in our test comparison, the performance obtained using the highest resolution triangle mesh still enables interactive frame rates. These results show that half floating point precision can be used to speed up computations without compromising the numerical accuracy.

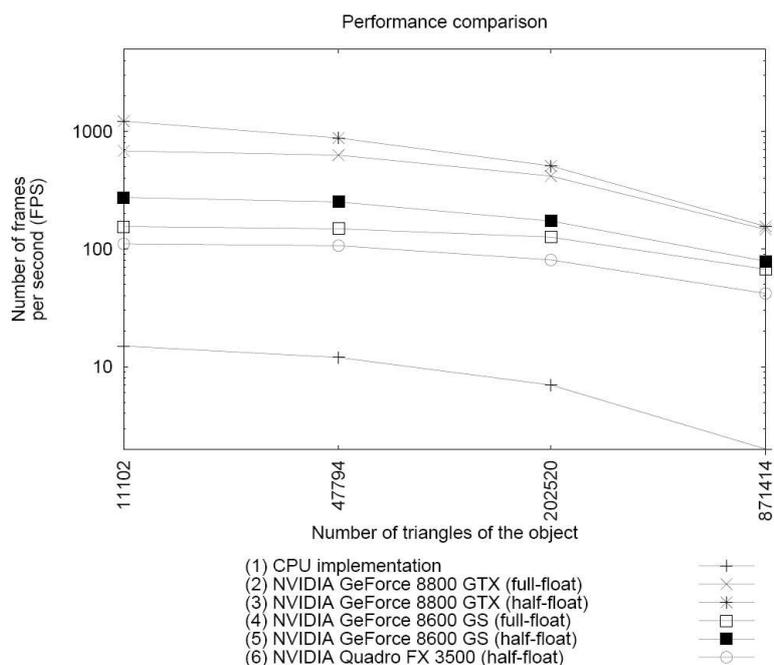


Fig. 7. Number of radiographs (1024 x 768 pixels) computed in one second.

4. Conclusion

We have presented here a simulator to perform PTC. The contents have been dictated by a task analysis. It includes real-time respiratory motion with two independent parameters (rib kinematics and diaphragm action), on-line fluoroscopy implemented on the GPU and haptic feedback to feel the soft-tissue behaviour of the organs during the needle insertion.

The next goal of our work is to provide feedback to the operator which is derived from use of performance metrics. Regions of interest such as no-go areas will be added to perform this quantitative analysis. Soft-tissue deformation due to the needle will be added to the simulation to replicate what is observed on the fluoroscopic display. Finally, this simulator will be extended to other interventional radiology needle insertion procedures by incorporating ultrasound rendering, also based on the GPU parallelisation.

Acknowledgement.

This report/article presents independent research commissioned by the National Institute for Health Research (NIHR). The views expressed in this publication are those of the authors and not necessarily those of the NHS, the NIHR or the Department of Health. The authors would like to thank Dr N. Freud and Dr J. M. Létang for their helpful comments about X-ray modelling and for providing the data necessary for the validation of our X-ray simulation code.

5. References

- 1Stern, J., Zeltser, I. S., Pearle, M. S. Percutaneous renal access simulators. *Journal of endourology / Endourological Society* 2007;21(3):270-3.
- 2Militello, L. G., & Hutton, R. J. B. (1998). Applied cognitive task analysis (ACTA): a practitioner's toolkit for understanding cognitive task demands. *Ergonomics*, 41 (11), 1618-1641.
- 3Grunwald, T., Clark, D., Fisher, S. S., McLaughlin, M., Narayanan, S., & Piepol, D. (2004). Using Cognitive Task Analysis to Facilitate Collaboration in Development of Simulator to

- Accelerate Surgical Training. Proceedings of the 12th Medicine Meets Virtual Reality Conference, 114-120
- 4Velmahos, G. C., Toutouzas, M. D., Sillin, L. F., Clark, R. E., Theodorou, D., & Maupin, F. (2004). Cognitive task analysis for teaching technical skills in an inanimate surgical skills laboratory. *The American Journal of Surgery*, 187 (1); 114-119.
- 5Reynolds, R.T., & Brannick, M. (2000). Is job analysis doing the job: extending job analysis with cognitive task analysis. *The Industrial Psychologist*, 39 (1), 63 – 67.
- 6McMinn R. M. H.: *Last's Anatomy: Regional and Applied*. Churchill Livingstone, 1990. ISBN: 0-443-03483-4.
- 7Wilson T. A., Legrand A., Gevenois P.-A., Troyer A.: Respiratory effects of the external and internal intercostal muscles in humans. *J. Physiol. (Lond.)* 530, 2 (2001), 319-330.
- 8Li Y., Brodlić K.: Soft object modelling with generalised chainmail - extending the boundaries of webbased graphics. *Comput. Graph. Forum* 22, 4 (2003), 717-727.
- 9N. Freud, P. Duvauchelle, J. M. Létang, D. Babot, Fast and robust ray casting algorithms for virtual X-ray imaging, *Nuclear Instruments and Methods in Physics Research B* 248 (1) (2006) 175–180.
- 10Ruspini, D., Kolarov, K. And Khatib, O. 1997. "The Haptic Display of Complex Graphical Environments". *Computer Graphics Annual Conference Series*, 345-352.
- 11Maurin, B., Bayle, Z., Gangloff, De Mathelin, Gangi, S., And Forgione. "In Vivo Study of Forces During Needle Insertions". *Proceedings of the Medical Robotics, Navigation and Visualisation Scientific Workshop 2004*, 1-8. Remagen, Germany, March 2004.