

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Development of a Medical Training System with Integration of Users' Skills Assessment

Ronei M. Moraes and Liliane S. Machado
Federal University of Paraíba
Brazil

1. Introduction

Virtual Reality (VR) joins several areas to produce environments that contain elements with realistic behaviour. The real-time processing of users interactions is a mandatory requirement and one or more user senses can be explored in VR systems. Machado & Moraes (2010) presented the multidisciplinary aspect of the VR systems and pointed out their importance for training.

Particularly, the training of medical procedures can be improved by VR systems. It occurs due to the fact that traditional training in medicine is frequently performed with cadavers or guinea pigs, which do not present the same characteristics of live bodies. Practice supervised by a tutor, with real patients, is another usual situation used for training. Both cases present problems: dead bodies, besides the degradation along the time, cannot be reused if cuts or punctures were done and need to be discarded after some manipulation; and supervised training can be risky or even uncomfortable for patients. Additionally, supervised training depends directly on the cases attended, what could not represent the variability and diversity of cases that students must learn. These observations make VR powerful to improve training, since it can rebuild digitally the same components of real procedures and can present virtual bodies with similar features of live bodies. Moreover, it can allow realistic and interactive simulation, in safe conditions, for users and patients; can provide variability of cases and offers reusability of the structures.

A very important aspect related to VR for medical simulation is the possibility of monitoring users' actions. Since VR systems are computational applications, all input and output data can be stored and used to assess users. This feature allows users to know their skills and to identify the points that must be improved in the execution of the procedure. This is particularly important in blind procedures in which a videotape of the execution cannot give possibilities of assessment by an expert. However, only recently this resource has been explored (Machado et al., 2000; Machado & Moraes, 2010).

This chapter will present the development process of a medical training system for bone marrow harvest procedure, a minimally invasive procedure. The problem approached will be described in order to identify the senses involved in the real procedure. Then, difficulties to assess trainees' skills and problems present in a real procedure, such as risks for donors, will be discussed allowing the identification of key points in the conception of a simulator based on VR. Those key points, the project budget and technological limitations will help to define the main features to be included in the computational architecture of the simulator.

The use of a framework that integrates several tools and methods will be introduced to demonstrate how this kind of software can decrease implementation time and standardize the development of this kind of application.

2. Development difficulties

There are four aspects related to the development of simulations for medical training: know the procedure, presence of a multidisciplinary team, presence of methods to assess the quality of training performed in the simulator and knowledge of computational tools and technologies for programming the application. It makes challenging the activities of the researchers to produce advances that could benefit all areas involved, without duplicity of efforts.

Know the procedure refers to obtain a detailed description about all elements involved in it: objects, structures, tools, anatomical parts, ways to perform the procedure and its variations, senses included (sight, touch, hearing, smell), range of expected time necessary to perform the procedure, and other aspects necessary to have a complete description. That information should be analysed according to the project budget and the technology available and can demand changes in the approach to be adopted. However, any decision must be discussed among the team: computer scientists and engineers will point out the technological possibilities and physicians will evaluate the approaches proposed, as example. The method to be used to assess users in the simulator must also be discussed according to the simulator subject. The detailed description will provide information that will allow identifying the type of variables included in the procedure. It must be discussed with a statistician or mathematician to define an adequate and good assessment method capable to deal with input and output data of simulator. During the implementation and after it, the development team will continuously refine and correct the approach adopted. Finally, the physicians must do the validation of the simulator.

A simulator based on VR must contain the same steps present in a real procedure and can use volumetric models, stereoscopic visualization, haptic systems, deformation methods and interactive cut. The use of all those features requires a lot of processing and can compromise the real-time performance. Thus, a VR simulator for medicine is defined and developed to deal with specific parts of the human body and present some limitations: according to the procedure requirements, some VR techniques are more explored than others. The architecture components could present educational aspects beyond training. Educational contents include explanations of the procedure, models with transparent view, attach, detach and other 3D manipulation tools, that only can be present by a VR application, to help users to solve some doubts about the procedure. Additionally, data collected during interaction in a VR-based simulation can be used to assess users' performance in training and to allow them to identify mistakes and improve their skills. An online assessment tool coupled to the simulator can do it.

A difficulty related to the development of simulators, which explore more than one human sense, is the integration of the tasks related to each sense and their synchronization. Most of the simulators found in literature were developed through the implementation and integration of all routines necessary. Because there are several devices and specific APIs (Application Programming Interface) dedicated to their programming, a programmer with experience to deal with each device or API is necessary to develop the VR simulators. Methods for collision detection, deformation, and stereoscopic visualization also need to be

completely implemented each time an application is developed. Besides, the several methods need to deal with the same models and interaction devices. Then, all this information and all application tasks must be synchronized to avoid delays or inconsistencies that can compromise the realism. Figure 1 presents a diagram with some tasks that can be present in a simulator for medical training: the tasks can depend on process from other tasks. Finally, there isn't any guarantee of compatibility among simulators and codes are not usually shared among groups, fact that causes constant duplicity of efforts and recoding of the same tasks.

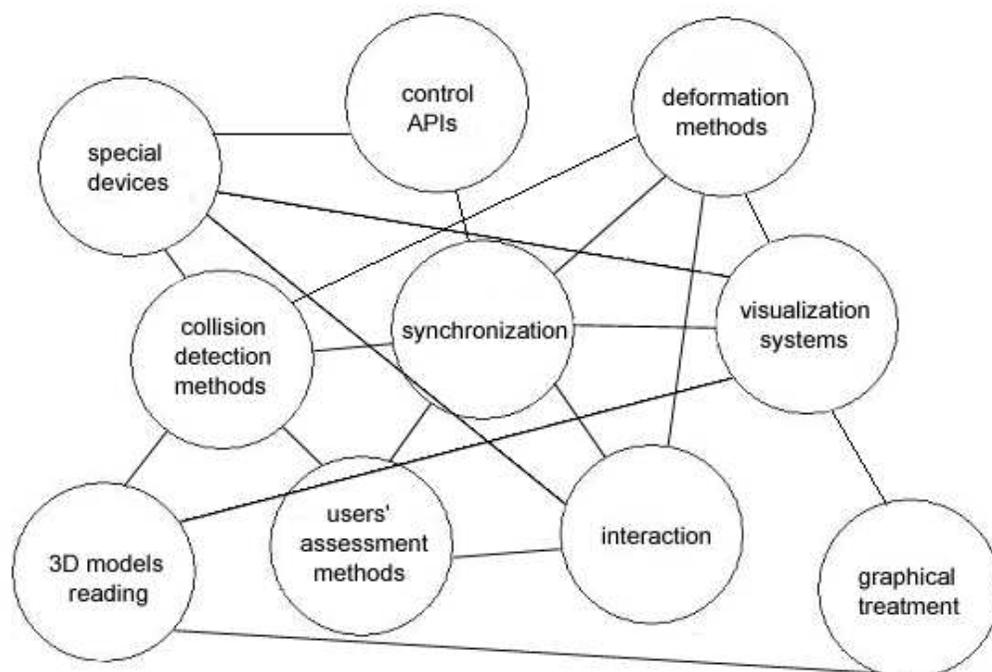


Fig. 1. Tasks that could compose medical simulators based on VR and their relation.

2.1 The framework CyberMed

To programming a simulator it is necessary to know a set of methods and tools: the method for generation of stereoscopic images and their adequacy to the visualization device chosen must be considered; the haptic device has specific API for programming; there are different methods for collision detection between objects in virtual environments; deformation methods depends on collision detection and can be more or less complex, depending on the one chosen; assessment is related to the type of input data (interaction) and demand the knowledge of mathematical or statistical concepts; the models must be readable; and, finally, all processes must be synchronized.

The use of frameworks was recently adopted to minimize duplicity of efforts, to make interoperable the simulators and to allow fast development. It means that the stage of implementation can be accelerated if programmers do not need to learn and develop all methods and tools related to each task. Examples of frameworks for the development of medical simulators are the GiPSi, Spring, SOFA, SSVE, Chai3D, ViMeT, and CyberMed (Machado et al., 2009; Correa et al., 2009). In Table 1 is possible to observe that most of them are open source, feature that augments the possibilities of discussion and code exchange among programmers. Support to haptics is also an important component since it allows

including routines that provide sense of touch and force feedback for users. However, only the CyberMed presents support to assessment and can be pointed as a framework for the development of training simulator capable to provide feedback about users' skills.

Framework	Open Source	Free License	Support to haptics	Support to assessment
GiPSi	X	X	X	
SSVE				
Spring	X		X	
SOFA	X	X		
CHAI3D	X	X	X	
ViMeT	X	X	X	
CyberMed	X	X	X	X

Table 1. Comparison among frameworks for medical development based on VR.

CyberMed is a framework to the development of VR simulators based on VR for commodity computers (Souza et al., 2007). It is composed by a set of open and free libraries that allows integrating new methods or supporting new devices. It also supports low cost devices as much as high-end devices, depending on the budget of the project. This flexibility permits to attend a large group of applications and programmers can select the resources available according to the goal and public of the final system. It allows programming in high-level but permit expanding its functionalities in a low-level programming. It is available for download at <http://cybermed.sourceforge.net/>. Figure 2 presents CyberMed layers and their disposition inside the system.

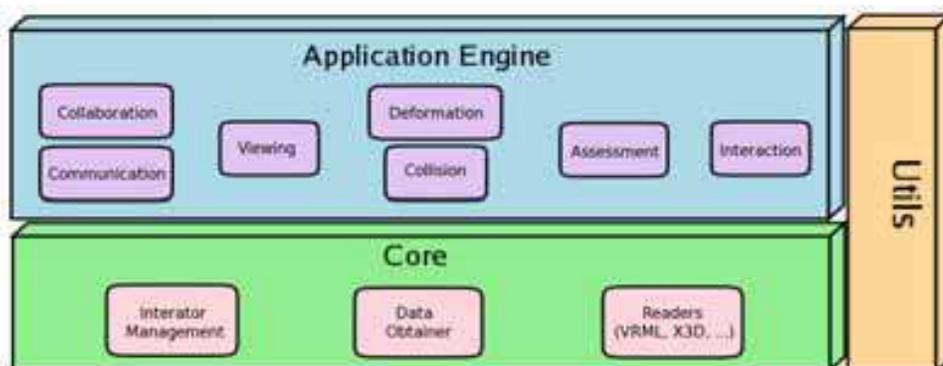


Fig. 2. CyberMed architecture.

The architecture of CyberMed is based on the object-oriented paradigm and follows some pattern designs to guarantee software reuse and agile development. Then, *singleton* and *Abstract Factory* were patterns applied in CyberMed development: *singleton* offers a single instance of a class to allow singleness of information; and *Abstract Factory* allows the creation of interfaces to make new objects with common operations.

The framework is composed by a set of layers that provide services for its users. Each layer abstracts several concepts to facilitate the development of VR applications. Nowadays, CyberMed offers support to: three different modes of stereoscopy, mass-spring deformation, two collision detection modes, import 3D meshes, optical and magnetic tracking, haptics, network collaboration, synchronization, and three different assessment methods.

3. Design of the simulator

The design of the simulator starts with the complete description of the procedure. The bone marrow transplant is a minimally invasive medical procedure to treat recently considered incurable diseases, as some specific occurrences of cancer. It is used to replace the cells of the immune system, mainly after chemotherapy or radiotherapy sessions. To obtain bone marrow it is necessary to extract it through many material aspirations from the iliac crest bone marrow (sometimes it includes the sternum bone also) of a donator under general anaesthesia. The harvest is executed blindly without any visual feedback except the external view of the donor's body. So, the physician needs to touch patient's body and identify the position of the iliac crest. After, he will insert a needle from this position to reach the bone marrow. In order to do that, he must apply adequate force and feel the tissue layers pierced by the needle.

The knowledge of the anatomic structures of the pelvic region, as well as the physician ability to find the iliac crest under the skin and the perception of the tissues pierced by the needle are essential to perform the bone marrow harvest procedure. Thus, it is possible to observe that the sense of touch is essential to perform the procedure and will be from it that the physician will decide the correct position to insert the needle and when to stop doing it. The dexterity of the physician will determine his skills.

The process of learning the bone marrow harvest procedure can be divided in three stages: understand the anatomy of the pelvis, learn how to identify the iliac crest and acquire dexterity to insert the needle, with correct force and precision, to reach the bone marrow. The elements necessary to compose these stages are: anatomical models of structures of the iliac region; needle for harvest; hand or fingers for palpation; hardness and elasticity properties of tissues from each structure; colour of structures.

Factors that determine a correct execution are related to the time of harvest, correctness of place in needle insertion, and adequate application of forces during the piercing. This information can be used for decision making about users dexterity and their skill level.

3.1 Assessment methodologies

The research area on training assessment for simulators based on VR is recent (Burdea & Coiffet, 2003). The early works in that area probably were proposed by Dinsmore et al. (1996; 1997) (Langrana et al., 1997) that used a quiz to assess users of a VR environment to identify subcutaneous tumors. The quiz contained questions related to the diagnosis and hardness of tumor. Similarly, Wilson et al. (1997) created a minimally invasive system (MIST) in which each task could be programmed for different difficulty levels. Performance data of each user could be saved to post analysis (offline) by an expert or statistical methods.

In parallel, methods to assess surgical skills have been developed by several research groups. Some of them use statistical models to do that offline (Derossis et al., 1998) and others use statistical methods to show that through VR based systems is possible to discriminate between expert and novice physicians (Taffinifer et al., 1998; Gallagher et al., 2001). It was showed also that surgeons trained in VR systems could obtain better results (Gallagher et al., 1999) when compared to others trained by traditional methods. Additionally, the assessment of psychomotor skills in VR systems that include haptic devices can quantify surgical dexterity with objective metrics (Darzi et al., 1999). Thus, VR systems for training can be used to provide metrics to a proficiency criterion of learning (Darzi et al., 1999; Gallagher et al., 2005). Due to those reasons, McCloy & Stone (2001) pointed out the assessment of psychomotor skills as the future of medical teaching and training.

The first proposal for online training assessment in VR systems was presented by (Burdea et al., 1998) and was based on a boolean logic that compared diagnoses provided by users with correct ones stored in the simulator. However, ordinary computers of that generation were not able to run simultaneously virtual reality environments/ simulators and online assessment systems if several interaction variables were monitored. After that, more sophisticated assessment methods were proposed for several training systems (McBeth et al., 2002; Huang et al., 2005; Mackel et al., 2006; Kumagai et al., 2007; Farber et al., 2008).

Apart from methods for evaluating a user, have also been proposed methodologies for evaluation of multiple users interacting in the same VR environment (Moraes & Machado, 2007). It is important to note that there are evaluations systems for training systems based on Web (Moraes & Machado, 2005b; 2006; Machado & Moraes, 2006). Some methodologies, which cannot be able to run in real time in commodity computers, can be implemented using embedded systems, coupled to them (Moraes & Machado, 2008).

The choice of an assessment methodology is strongly related to the procedure itself. For example, in a procedure in which information can be obtained from procedures performed in simulators based on VR are fully quantitative, numerical methods based on probability distributions can be most appropriate (Machado & Moraes, 2006). However, one should take into consideration what types of such information specialists are able to interpret. In cases where such information coming from specialists are imprecise or vague, modelling from fuzzy sets (Moraes & Machado, 2009a) or hybrid modelling can be good choices (Moraes & Machado, 2010). Can still be used in knowledge-based modelling, such as rule-based expert systems (classical or fuzzy) (Machado et al., 2000). Approaches based on models for discrete variables or statistical models based on binomial or multinomial distributions are not indicated, since the need for changes on the original variables that can cause loss of information (Moraes & Machado, 2009b). In cases where information is qualitative, the use of expert systems can be a good solution (Machado et al., 2008). Other approaches based on multinomial probability models (Moraes & Machado, 2009a; 2009b) can also be used for these cases. There are still cases in which information is obtained from both qualitative and quantitative variables. In these cases, evaluation method to be used should take into consideration these aspects and be able to analyse the two types variables simultaneously (Machado & Moraes, 2009).

Because users can have forgotten some decisions and actions done during the training, the main problem in this kind of evaluation is the interval between training and evaluation. Additionally, some performances cannot be classified as good or bad due to procedure complexity. Thus, the existence of an evaluation system coupled to a simulation allows fast feedback that can be used improve user performance. However, it demands low computational complexity in order to not compromise the simulation performance and high accuracy to do not compromise the quality of evaluation. Figure 3 shows the independence between simulator and evaluation systems in spite of their simultaneous action.

Theoretically, the system can collect any information related to position, forces, torque, resistance, velocity, acceleration, temperature, angle of visualization, sound, smells, temperature, velocity and shape. Spatial position can be captured by tracking devices attached to some part of users' body or to an object manipulated by them. Those devices can collect movements in three or more degrees of freedom in constant intervals of time. Such information allows VR system to calculate object position in the virtual scene, the velocity of the movement and its acceleration. In a similar way, haptic devices (Burdea & Coiffet, 2003) also can do the same, but in a limited physical space. In both examples, sampling rates vary

according to the device chosen. The positioning of objects manipulated by users can allow the identification of collision with other objects of the virtual scene, besides the direction of contact and its intensity. In order to achieve this, position vectors with last and present position are used to determine the collision velocity and intensity.

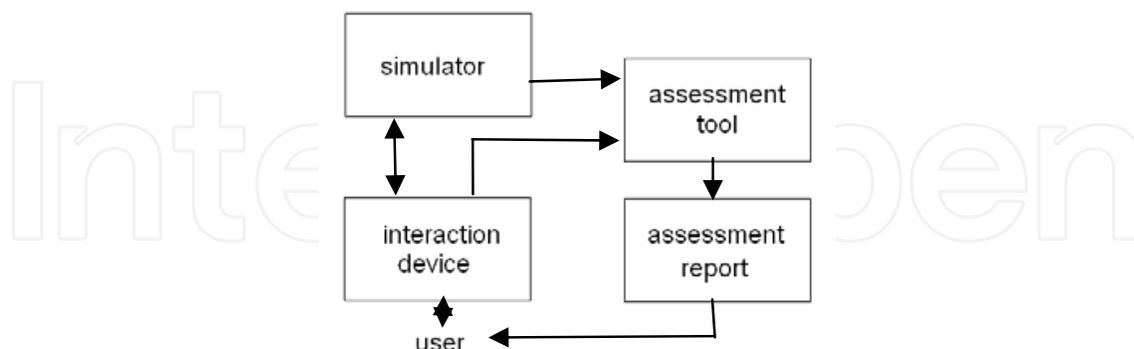


Fig. 3. Assessment and its relation with VR simulator.

Reactions from data processing collected from interaction can be send to users as force feedback, a new point of view of the scene, deformation of objects, sound, increase or decrease of temperature, change of objects (as cuts or divisions) and smells. The intensity of feedback will depend on device capabilities. However, the continuity of reaction is an important factor in the levels of immersion.

a. Maximum Likelihood

Maximum Likelihood decision rule is the most common statistical technique used in data classification. Its application for training evaluation based on VR was presented by Moraes & Machado (2005a). Formally, the classes of performance for an user are done by: w_i , $i=1, \dots, M$, where M is the total number of classes of performance. It is possible to determine the most probable class of a vector of training data X , by conditional probabilities (Johnson & Wichern, 2001):

$$P(w_i | X), \text{ where } i=1, \dots, M. \quad (1)$$

The probability done by (1) gives the likelihood that for a data vector X , the correct class is w_i . Classification rule is performed according to

$$X \in w_i \text{ if } P(w_i | X) > P(w_j | X) \text{ for all } i \neq j. \quad (2)$$

However, all the probabilities done by (1) are unknown. So, probabilities denoted by $P(X | w_i)$ can be estimated if there is sufficient information available for each class of performance. Using the Bayes Theorem:

$$P(w_i | X) = [P(X | w_i) P(w_i)] / P(X), \quad (3)$$

where $P(X) = \sum_{i=1}^M P(X | w_i) P(w_i)$.

As $P(X)$ is the same for all classes w_i , then it is not relevant for data classification. In Bayesian theory, $P(w_i)$ is called *a priori* probability for w_i and $P(w_i | X)$ is *a posteriori* probability for w_i where X is known. Then, the classification rule done by (2) is modified:

$$X \in w_i \text{ if } P(w_i | X) P(w_i) > P(w_j | X) P(w_j) \text{ for all } i \neq j. \quad (4)$$

Equation (4) is known as the maximum likelihood decision rule or maximum likelihood classification. However, it can be convenient to use (Johnson & Wichern, 2001):

$$g(X) = \ln [P(X | w_i) P(w_i)] = \ln [P(X | w_i)] + \ln [P(w_i)] \quad (5)$$

where $g(X)$ is known as the log-likelihood function and it is known as discriminant function. Equation (5) can be used to modify the formulation done by (4):

$$X \in w_i \text{ if } g_i(X) > g_j(X) \text{ for all } i \neq j. \quad (6)$$

It is important to note that if statistical distribution of training data can assume multivariate Gaussian distribution, the use of (6) has interesting computational properties (Johnson & Wichern, 2001). If training data cannot assume that distribution, (6) can provide a significant reduction of computational cost of implementation. Thus, the user's interactions with the system are monitored and the information is sent to the evaluation system, where Maximum Likelihood is in its kernel. This method can produce good results even with small bases calibration. However it is necessary that these data are able to translate the scope of the problem addressed.

For the calibration of the evaluation system, an expert executes several times the procedure, labelling each one into a class of performance among M classes available. For each one, the vectors of each variable are storage. At the end of calibration, the information of variability about each one procedure class is acquired by calculate the mean vector and covariance matrix. These parameters are sufficient to characterize the multivariate Gaussian distribution for each class of performance. When it is done, the evaluation system is ready to run.

The users perform their training in the VR simulator and the Evaluation Tool based on Maximum Likelihood collects the data from that manipulation. All probabilities of data for each class of performance are calculated by evaluation tool and a w_j class of performance is assigned to the user at the end of simulation, according (4) or (6), which is reported to the trainee.

b. General Bayesian Network (GBN)

A Bayesian network is a probabilistic model that can represent a set of probabilities distributions from all variables in a complex process and also establish their relationships (Neapolitan, 2003). Formally, a Bayesian network is defined as directed acyclic graphs, denoted by G and a probabilistic distribution denoted by P . The graph G is a set of nodes and oriented arcs, where nodes represent variables in process and oriented arcs encode conditional dependencies between variables (Neapolitan, 2003). The dependencies are modeled by specific conditional probabilistic distributions (Krause, 1998).

Cheng & Greiner (2001) proposed a classification for Bayesian networks, according to their graph structure, in five classes: Naive-Bayes, Tree Augmented Naive-Bayes, Augmented Naive-Bayes, Bayesian Multi-net nets e General Bayesian Networks (GBNs). The choice of a specific structure to knowledge representation depends on dependencies relationship between variables, which describe that process. That choice is critical, because it changes the final results. The GBN is a generalized form of Bayesian networks, which allows nodes to form an arbitrary graph, rather than just a tree. Another important characteristic is that each child node cannot be connected to the final classes of evaluation.

The General Bayesian Network is convenient to serve as base for a training evaluation due to its generality. In that network, the dependencies between nodes can adjust itself to real

dependencies. Thus, it is possible to verify dependencies between variables during network modelling and put them in structure nodes of GBN, which did not occur in other structures. Formally, let the classes of performance be in space of decision Ω with M classes of performance. Let be $w_j, j \in \Omega$ the class of performance for a user and $X_i, 1 \leq i \leq n$, represents a node in GBN with n as the number of nodes in a graph. The joint probability distribution in GBN for an event is done by:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | X_{i-1}, X_{i-2}, \dots, X_1) \quad (7)$$

where $P(X_1, X_2, \dots, X_n)$ is the joint probability distribution and $P(X_n | X_{n-1}, X_{n-2}, \dots, X_1)$ is the conditional probability of X_n conditioned by its predecessor nodes $X_{n-1}, X_{n-2}, \dots, X_1$.

If the conditional independence between variables is verified, this permits simplifications in (7). Then,

$$P(X_1, X_2, \dots, X_n) = P(X_1 | w_j) P(X_2 | w_j) P(X_3 | w_j) \quad (8)$$

The node probabilities are associated to probability distribution. For example, a node A can have a Gaussian distribution and a node B , which depends on A , can have a bivariate Gaussian distribution, with a mean vector and a covariance matrix (Johnson & Wichern, 2001).

The structure of GBN is learned from data, as well as the parameters of conditional probabilities. By the use of probabilities calculus is possible to find dependencies among nodes in a Bayesian network. If those dependencies are founding and, if is possible to assume Gaussian distribution for nodes, dependencies can be estimated using multivariate linear regression (Neapolitan, 2003). Scores are used to help the estimation of the final structure of GBN for each class of assessment. In a first moment, a network is created with all independent nodes and an initial score is calculated. Following, all combinations are searched and an arc is designed between two nodes to obtain an increment of initial score. Then, the parameters for that nodes set are re-estimated using linear regression. This cycle is repeated until the total network score could be less than a predetermined value or a fixed number of cycles. This methodology has the ability to obtain better results for larger calibration databases.

Previously, an expert calibrates the system, according M classes of performance. The information of variability about these procedures is acquired using GBN based method (Moraes et al., 2009). The users perform their training in the VR simulator and the Assessment Tool based on GBN collects the data from that manipulation. All probabilities of data for each class of performance are calculated by GBN and a w_j class of performance is assigned to the user at the end of simulation, according to (7) or (8), depending on the particular case. At the end of the training, the assessment system reports the classification to the trainee.

For the calibration of the evaluation system, an expert executes several times the procedure, labeling each one into a class of performance among M classes available. For each one, the vectors of each variable are storage. At the end of calibration, the structure of GBN is learned from data, as well as the parameters of conditional probabilities for each class of performance. Using probabilities calculus is possible to find dependencies among nodes in a Bayesian network. If those dependencies are founding and, if is possible to assume Gaussian distribution for nodes, dependencies can be estimated using multivariate linear regression (Neapolitan, 2003). Scores are used to help estimate the final structure of GBN for each class

of assessment. In a first moment a network is created with all independent nodes and an initial score is calculated. Next, all combinations are searched and an arc is designed between two nodes for which an increment of initial score is obtained. Then, the parameters for that nodes set are re-estimated using linear regression. This cycle is repeated until total network score is less than a predetermined value or a fixed number of cycles.

The users perform their training in the VR simulator and the Evaluation Tool based on GBN collects the data from that manipulation. All probabilities of data for each class of performance are calculated by evaluation tool and a w_j class of performance is assigned to the user at the end of simulation, according (7) or (8), which is reported to the trainee.

c. Expert system based on Fuzzy Rules

As it is possible that some variables in the training system do not present an exactly correspondence to the real world, some measures cannot be exact. Then we must use fuzzy sets to measure those variables (Dubois & Prade, 1980).

In classical set theory a set A of a universe X can be expressed by means of a membership function $\mu_A(x)$, with $\mu_A: X \rightarrow \{0,1\}$, where for a given $a \in A$, $\mu_A(a)=1$ and $\mu_A(a)=0$ respectively express the presence and absence of a in A . Mathematically:

$$\mu_A(x) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A \end{cases} \quad (9)$$

Zadeh (1965) introduced the fuzzy set theory in 1965. A fuzzy set or fuzzy subset is used to model an ill-known quantity. A fuzzy set A on X is characterized by its membership function $\mu_A: X \rightarrow [0,1]$. We say that a fuzzy set A of X is "precise" when $\exists c^* \in X$ such that $\mu_A(c^*)=1$ and $\forall c \neq c^*, \mu_A(c)=0$. A fuzzy set A will be said to be "crisp", when $\forall c \in X, \mu_A(c) \in \{0,1\}$.

The intersection and union of two fuzzy sets are performed through the use of *t-norm* and *t-conorm* operators respectively, which are commutative, associative and monotonic mappings from $[0,1] \rightarrow [0,1]$. Moreover, a *t-norm* Γ (respec. *t-conorm* \perp) has 1 (respec. 0) as neutral element (e. g.: $\Gamma = \min, \perp = \max$) (Dubois & Prade, 1988). Thus, we can define intersection and union of two fuzzy sets as:

The intersection of two fuzzy sets A and B , with membership functions $\mu_A(x)$ e $\mu_B(x)$ is a fuzzy set C with membership function given by:

$$C = A \cap B \Leftrightarrow \mu_C(x) = \Gamma\{\mu_A(x), \mu_B(x)\}, \forall x \in X. \quad (10)$$

The union of two fuzzy sets A and B , with membership functions $\mu_A(x)$ e $\mu_B(x)$ is a fuzzy set C with membership function given by:

$$C = A \cup B \Leftrightarrow \mu_C(x) = \perp\{\mu_A(x), \mu_B(x)\}, \forall x \in X. \quad (11)$$

The complement of a fuzzy set A in X , denoted by $\neg A$ is defined by:

$$\mu_{\neg A}(x) = n(\mu_A(x)), \forall x \in X. \quad (12)$$

where: $n: [0,1] \rightarrow [0,1]$ is a negation operator which satisfies the following properties:

- $n(0)=1$ and $n(1)=0$
- $n(a) \leq n(b)$ if $a > b$
- $n(n(a))=a, \forall x \in [0,1]$

and a negation is a strict negation if it is continuous and satisfies

- $n(a) < n(b)$ if $a > b$.

The main negation operator which satisfies these four conditions is $n(a) = 1 - a$.

The implication function between two fuzzy sets A and B , with membership functions $\mu_A(x)$ e $\mu_B(x)$, is a fuzzy set C with membership function given by:

$$C = A \Rightarrow B \Leftrightarrow \mu_C(x, y) = \nabla\{\mu_A(x), \mu_B(y)\}, \forall x \in X, \forall y \in Y \quad (13)$$

where $\nabla: [0, 1]^2 \rightarrow [0, 1]$ is an implication operator which obeys the following properties: $\forall a, a', b, b' \in [0, 1]$:

- If $b \leq b'$ then $\nabla(a, b) \leq \nabla(a, b')$;
- $\nabla(0, b) = 1$;
- $\nabla(1, b) = b$.

The pure implications obeys too:

- If $a \leq a'$ then $\nabla(a, b) \geq \nabla(a', b)$;
- $\nabla(a, \nabla(b, c)) = \nabla(b, \nabla(a, c))$.

Expert systems (Rich & Knight, 1993) use the knowledge of an expert in a given specific domain to answer non-trivial questions about that domain. For example, an expert system for image classification would use knowledge about the characteristics of the classes present in a given region of the image to classify a pixel of that region. This knowledge also includes the "how to do" methods used by the human expert. Usually, the knowledge in an expert system is represented by rules in the form:

IF <condition> THEN <conclusion>.

Most rule-based expert systems allow the use of connectives AND or OR in the premise of a rule and of connective AND in the conclusion. From rules and facts, new facts will be obtained through an inference process.

In several cases, we do not have precise information about conditions or conclusions, then the knowledge in the rules cannot be expressed in a precise manner. Thus, it can be interesting to use a fuzzy rule-based expert system (Zadeh, 1988). However, it is important to emphasize that obtaining rules from an expert can be a long process and eventually must be done in several stages. This is the major limitation of this methodology.

For example, using a fuzzy expert system coupled to a surgical simulator based on virtual reality, experts can define, in an imprecise way, regions of tissue and bones in where is possible to find bone marrow harvest. To evaluate a user, values of variables are collected by a haptic device and sent to expert system for evaluation. The system must analyse each step performed by users and classify users, at the end of training, according to predefined performance classes. Machado et al. (2000) used five performance classes, described by the following fuzzy sets: *you need much more training*, *you need more training*, *you need training*, *your training is good* or *your training is excellent*, to identify if more training of bone marrow harvest procedure is or not necessary. The fuzzy rules of expert system are modelled by membership functions according to specifications of experts. Several types of membership functions can be used as trapezoidal, triangular and pi-functions and the fuzzy inference system used is Mamdani-type (Mamdani & Assilian, 1975). An example of rule for this expert system is:

IF Position_x is left_center AND Position_y is up_center
 AND Position_needle is acceptable AND Marrow_harvest is yes
 THEN Trainee_class is you_need_training

where: Position_x, Position_y are coordinates which the needle touch the patient body; Position_needle is the angle of needle input to body of patient; Marrow_harvest shows the success or failure of trainee to harvest bone marrow and Trainee_class is the classification of trainee.

In a different way from previous methods, the calibration of an expert system based on fuzzy rules is made by extraction of knowledge from one or several experts. In general, the experts explain the procedure and which variables must be taking in account in evaluation process. From those variables is necessary to know their parameters (which translate by fuzzy numbers) and as they are related with evaluation process to create rules to express each one of the M classes of performance. The number of rules depends on the complexity of procedure. When users perform their training, the Evaluation Tool based on Expert System collects the data from that manipulation. All rules are applied over data and, according to the expert system, one class of performance w_j is assigned to the user at the end of simulation, which is reported to the user.

4. Implementation

A simulator based on VR must contain the same steps present in a real procedure and can use volumetric models, stereoscopic visualization, haptic systems, deformation methods and interactive cut. The use of all those features requires a lot of processing and can compromise the real-time performance. Thus, a VR simulator for medicine is defined and developed to deal with specific parts of the human body and present some limitations: according to the procedure requirements, some VR techniques are more explored than others. The architecture components could present educational aspects beyond training. Educational contents include explanations of the procedure, models with transparent view, attach, detach and other 3D manipulation tools, that only can be present by a VR application, to help users to solve some doubts about the procedure. Additionally, data collected during interaction in a VR-based simulation can be used to assess users' performance in training and to allow them to identify mistakes and improve their skills. An online assessment tool coupled to the simulator can do it.

In order to deal with the main points of the learning of bone marrow harvest (BMH), the three stages described in the design were considered. It was also decided that learning and practice should be treated in different moments, and the simulator was conceived to present these two possibilities. When is learning the anatomy and understanding the procedure, the user should be able to view internally the virtual body and become familiarized with the tools and tissues properties. However, the practice should reproduce exactly the reality of the procedure and the user must be assessed in this process. Figure 4 presents how the different moments were approached in the simulator and the tasks related to each one.

Sight and touch are the two senses identified as fundamental in the procedure. The touch is necessary for the palpation of the virtual body and for the puncture with the needle. In this case, the use of a haptic device is mandatory to simulate the tissues properties and provide touch feedback for users. Haptic devices are special devices that incorporate sensors and

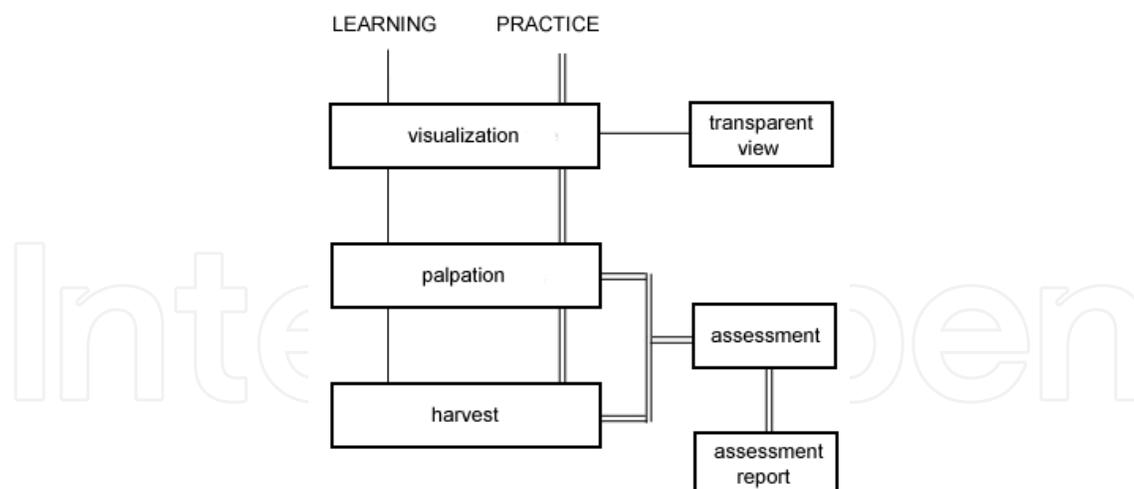


Fig. 4. Learning (I) and practice (II) tasks defined for the bone marrow harvest simulator.

actuators to capture user actions and provide touch and force feedback. The perception of touch allows identifying texture of objects and force feedback allows feeling sense of position and movement associated to the interaction with objects. Haptic devices with 6DOF¹ permit spatial movements and are the type of device necessary to the simulator for BMH. If the manipulation will occur in three dimensions, the visualization must follow the same principle and stereoscopic visualization will also be necessary. Figure 5 presents the haptic and visualization devices selected for BMH simulator.



Fig. 5. 6DOF haptic device (left) and shutter glasses for stereoscopic visualization (right).

Five models compose the application: three for the human body structures and two for the interaction object. The interaction objects, finger and needle, were modelled with the Blender package and saved in VRML 2.0 format. The models of human body structure were obtained from a previous work (Machado, 2003) and represent the skin of the pelvic region, the iliac bone and the bone marrow (Figure 6).

The tissues properties are important information for the simulator. They are used to give to models the same properties of the real tissues. In this case, textures can be acquired or designed, and colours must be defined. Touch properties, as hardness, elasticity and roughness must also be acquired. Because several tissues do not have a numerical value to describe their properties, it is common to acquire them from physicians' impressions. A haptic calibrator was used to do that (Figure 7). Then, some virtual spheres were presented to

¹ DOF = degrees of freedom

physicians and they could select and refine their properties according to what they feel when manipulate the real tissues: first they do for it for skin, after for bone and, finally, for bone marrow. The properties they could calibrate were: hardness, roughness, friction and viscosity.

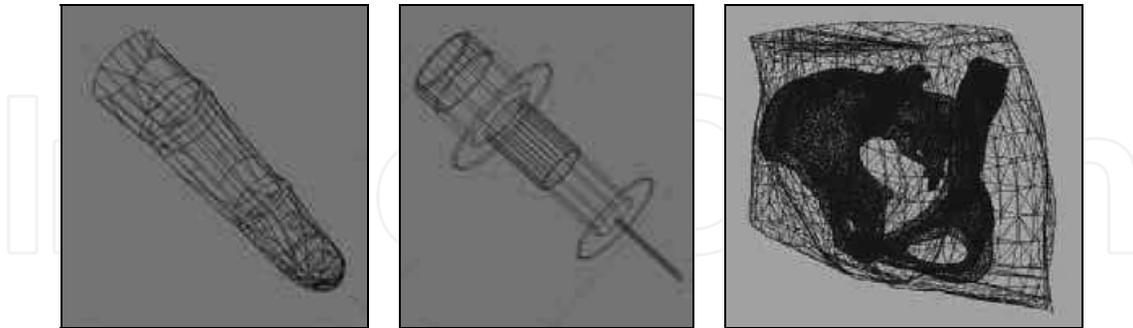


Fig. 6. Models conceived to represent the interaction device and pelvic structures (skin, iliac bone and bone marrow).



Fig. 7. Tool for calibration of material properties from physicians' impressions.

There is a last information necessary to make the simulator and it refers to the correct and wrong ways to perform the bone marrow harvest. However, describe it in natural language with details is not an easy tasks, even for physicians. Thus, the best way to do that will be making the physicians use the simulator and perform the procedure in several ways labelling what is correct and what is not correct. In this case, their interactions can be saved and used to construct models for users' assessment.

4.1 Coding with CyberMed

The application was implemented using the follow CyberMed classes: CybView, CybMouse, CybParameters, OF, CybInterator, CybHaptics and CybAssess. These classes offer the functionalities, respectively: visualization, mouse interaction, storage and management of 3D models, interaction control, haptic control and online user's assessment.

The five tasks defined in the design process were implemented. The CybView class allowed choosing the visualization mode of the application. In spite of CyberMed support four different view modes, the simulator was defined to provide only the monoscopic and temporal multiplexing visualization modes (using shutter glasses). With this class two light

sources were defined and an illumination model was enable to the visual scene and its objects. The menu entries were also made in the CybView. The objects used in the visualization are obtained from the OF and CybParameters classes which store the models topology, the model used to represent the interaction device and the transformation matrixes of the scene. The CybInterator class was used to choose the interaction object. The CybAssess will access in real-time the transformation matrixes and CybInterator to the user's assessment. In order to reach that, it was used the CybAssess with an assessment method based on Maximum Likelihood. The CybHaptics class allowed setting the material properties of the models and enabling the haptic device. This class provided an abstract layer for several API used to programming haptic devices and did not demand knowledge about the device API.

Figure 8 shows parts of the implementation with the CyberMed used to read the models and set the material properties for the haptic interaction. In that figure, it is possible to observe the reading of five VRML models in lines 1 to 8. The lines 10 to 16 define the properties of the first layer, the skin, as being touchable and pierceable. Observe that there isn't any mention of the haptic device in use. It is done in a single way with the class CybInterator and is independent of device manufacturer.

The visualization mode chosen does not need any special programming and the only thing necessary is instance the view mode and start it:

```
/* instance view mode for shutter glasses */
  CybViewShutter view;
/* Initialize visualization*/
  view.init();
```

```
1 int numLayer = 5;
2
3 // read the models
4 char *arqname[30] = {"1.wrl","2.wrl","3.wrl","4.wrl","5.wrl"};
5 ofMesh<cybTraits> malha[numLayer];
6 CybDataObtainer<cybTraits> data(numLayer);
7 ofWrlReader<cybTraits> entrada;
8 CybParameters cybCore;
9
10 // set haptic material properties for the skin
11 interator.hapticDevice.createHapticLayers(0,true);
12 interator.hapticDevice.createMaterialPropertyContext(numLayer,true);
13 interator.hapticDevice.setMaterialPropertyValue(0,POPTHROUGH,1.0f);
14 interator.hapticDevice.setMaterialFace(0,POPTHROUGH,FRONT);
15 interator.hapticDevice.setMaterialPropertyValue(0,DYNAMIC_FRICTION,0.6f);
16 interator.hapticDevice.setMaterialFace(0,DYNAMIC_FRICTION,FRONT);
17
18 // set haptic material properties for the bone
19 interator.hapticDevice.createHapticLayers(1,true);
20
21 // association of the contact point to a model point
22 interator.setObjectType(0,-0.020913,0.078050,-10.034132);
23
24 // enable material property for haptics
25 interator.hapticDevice.enableHapticMaterialProperty();
```

Fig. 8. Code of the BMH simulator using CyberMed 1.2.

The menu was implemented to be dynamically modified, according to user choices. A visual exploration is enable in the Study module and the user can choose which structures he wants to see and set their transparency. Figure 9 shows the Study module and the menu options available. The user can also modify the position and orientation of the structures through mouse interaction. If shutter glasses were available, it is possible to start the stereoscopy visualization.

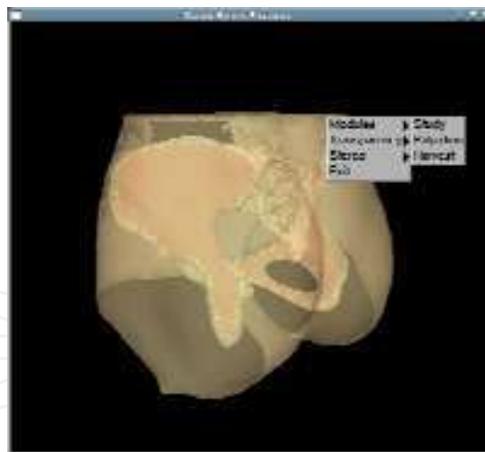


Fig. 9. Screenshot of the transparent view task and menu options available.

The second module available is the Palpation module. For this step of the training, all interaction with the objects was disabled and the position of the objects was fixed. In fact, the only visualization possible is from the backside of the skin model. Because the visualization of the bone and bone marrow models is not allowed, these models were disabled and will not be rendered. However, they can be identified by touch since the user starts the haptic interaction (Figure 10). With the haptic device the user will be able to feel the different material properties throughout the skin and identify a hardest area, located under the iliac crest (not visible). In this module, the finger model was related to the haptic device and the point of contact is located on its tip.



Fig. 10. Palpation tasks and the representation of the haptic device by a finger.

In the last module, the Harvest, the user can practice how to harvest the bone marrow. As in the Palpation module, movements with the body models are not allowed and a needle represents the haptic device. In this module is possible to penetrate into the models with the haptic device and all body structures - skin, bone and bone marrow - are haptically displayed. It allows reaching the bone marrow under the skin and inside the bone.

To the online user's assessment the simulation had to be executed several times to calibrate the assessment tool. This stage allows acquiring the assessment parameters to be used in the on-line assessment. In order to reach that, an expert executes several times the training and at the end of that, he/ she labels each one into M classes of performance (M=5 in the Figure

8: “Well Qualified”, “Qualified”, “Need Some Training”, “Need More Training” and “Novice”). This procedure is necessary to measure the statistical variability for each class of performance and improve the accuracy of the assessment method. It was used the Maximum Likelihood based method (Moraes & Machado, 2005a). Figure 11 shows the calibration stage of the simulator in the harvest module. The only difference between the calibration procedure for training assessment and the final training application is the presence of calibration options in the menu. In the calibration procedure, these options will be selected by an expert to label the class of performance for each calibration procedure. In the final application this menu option is replaced by an assessment option which can be selected by the user to receive his assessment report at the end of the simulation.

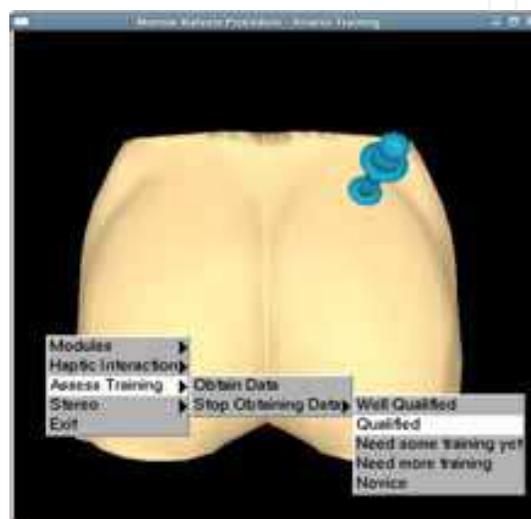


Fig. 11. Assess of training: acquisition of the assessment parameters.

5. Results

The bone marrow harvest simulator was designed and developed by Machado (2003). The present work used the same models and material property parameters of the previous work to develop a new application, with the same training goal but using a framework in its development. To compare the applications, the previous simulator will be called BMH Simulator and the new application CybBMH. Table 2 shows the main differences between BMH Simulator and CybBMH.

The CybBMH was developed only using free tools: GNU C++ language and the CyberMed system. The modeling package used to model the needle and the finger was the Blender, also free. Because the CyberMed integrates and synchronizes visual and haptic routines, it was not necessary the use of other tools for the development of the CybBMH. In opposite, the BMH Simulator required the use of several tools that must to be integrated, besides the necessity of synchronization of the visual and haptic tasks. The use of a single class to store the models topology allows utilizing the same three objects for visual and haptic scenes in the CybBMH. Additionally, two models were used to represent the haptic device in the visual scene. The illumination treatment offered by the CyberMed also allows providing a better visual quality. The visual quality of the simulators can be observed in the Figure 12. The BMH Simulator used three models for the visual scene and other three models for the haptic scene. Due to performance reasons, the haptic models were simplified and the

number of vertexes was reduced. Then, visual and haptic scenes did not share the same object and was necessary to use six different models in the past simulator. The representation of the haptic device was done through textures applied on crossed plans. In spite of optimizing the rendering process, this approach presented graphic imperfections.

	BMH Simulator	CybBMH
Operational system	Windows™	Linux
programming tools	- Visual C++ - OpenGL - Ghost API - Microsoft Foundation Classes	- GNU C++ - CyberMed
Haptic device	Phantom Desktop	any Phantom family device
Stereoscopy method	temporal multiplexing	temporal multiplexing
Number of models	6	5
Time for application development	~1 year	~2 months

Table 2. Main differences between the simulator previously developed and the new simulator.

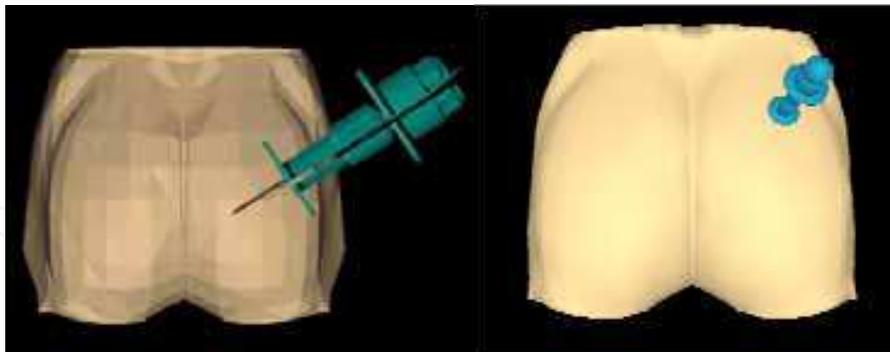


Fig. 12. Visual results of a previous application (left) and of the simulator developed with the CyberMed (right).

Both simulators support the use of shutter glasses and haptic devices. However, in the BMH Simulator the use of this hardware required a previous study of OpenGL functions to implement the routines for stereoscopic images generation. The same effort was necessary to integrate the haptic device and its routines to the application through the device API. In that simulator, the application window and menus were defined using the Visual C++ IDE (Integrated Development Environment) and the Microsoft Foundation Classes. The efforts needed to learn how to use all the tools mentioned above and to implement the

application demanded approximately one year for the complete development of the simulator. The CybBMH also demanded efforts to its development. However, it was focused in the use of the CyberMed classes. By the definition of objects and by the access to their methods, it was possible to use methods, already integrated, ready to deal with stereoscopy, haptic devices and menus. It allowed reducing the time of development of the simulator to approximately two months. Additionally, the CyberMed classes provided an on-line assessment method to assess user's performance. This functionality offers a performance feedback to the user and can be used to know his ability level.

The sequence of the steps of the simulation is the same in both applications. The dissection and the transparency settings available in the Study module can be equally performed through menus. However the illumination model provided by the CyberMed classes allowed to offer better graphic results in the CybBMH, especially in the Study module, due to the presence of three models composed by thousands of points (skin, bone and bone marrow).

Two haptic devices of the Phantom family (Sensable, 2010) were tested and used with the CybBMH: the Omni and the Phantom Desktop. The exchange of the haptic devices did not require any change in the final code of the application. The main differences between the two devices are the size of the workspace, the precision and the amount of force which can be exerted.

The time consumed in the development of the BMH Simulator and the CybBMH did not include the acquisition of the simulation parameters or the design of models used in the application.

5.1 Assessment of users

The framework CyberMed presents a higher level of abstraction, is dedicated to the development of medical applications and presents methods for on-line assess users' performance. In the version 2.0 of CyberMed, three evaluation methods are available: Maximum Likelihood, General Bayesian Network and Expert system based on Fuzzy Rules, which were described early. All of them can be applied to bone marrow harvest simulator.

When a training system based on VR is implemented, always need to involve experts in the area object of the system, not only in the design and calibration properties, but also to assess the final version of the system. The ideal way to evaluate is to perform procedures under various specific conditions in which the system is able to allow. So, the experts through questionnaires record their impressions about the system. Another option is doing usability tests with the experts. At this stage, some problems still can be detected and should be fixed, after which another round of evaluation is performed. Once the experts qualify the system, it really becomes useful for the medical community that will use it.

In the case of development of the bone marrow harvest system, experts followed its conception from the Children Hospital of the University of São Paulo (Brazil). They calibrated the system properties and tested them. At the end of development, system qualification tests were performed (Machado, 2003). When redesigning the system using CyberMed framework, these experiences have been incorporated and the configuration parameters were reused from the previous system (Souza et al., 2007).

For the problem of bone marrow harvest, it is possible to choose any one of three methods of assessment in CyberMed. However, in this case was chosen the method of maximum likelihood, due to limitation of other methods on the size of the database and the expert knowledge.

To verify performance of the method of maximum likelihood and for reasons of general performance of the VR simulator, the following variables were chosen to be monitored: spatial position, velocities, forces and time on each layer. Previously, an expert, according to M classes of performance defined by him, calibrated the system. The calibration process consists in to execute several times the procedure and to classify each one according to classes of performance. The number of classes of performance was defined as $M=3$: 1) correct procedures, 2) acceptable procedures, 3) badly executed procedures. So, the classes of performance for a trainee could be: "you are well qualified", "you need some training yet", "you need more training". Sixty samples of training (twenty of each class of performance) were used for calibration of evaluation system. After that, users performed 150 procedures. The information of variability about these procedures was acquired using the maximum likelihood method. In our case, we assume that the source of information for w_j classes is the vector of the sample data D. The user makes his/ her training in the virtual reality simulator and the evaluation tool collects data from his/ her manipulation. All probabilities of data for each class of performance are calculated and at the end the user is assigned to a w_j class of performance. So, when a trainee uses the system, their performance is compared with each expert's class of performance and the assessment tool assigns the most appropriated class, according to the trainee's performance. At the end of the training, the assessment system reports the classification to the trainee.

The Cohen's Kappa Coefficient was used to perform the comparison of the classification agreement, as recommended in the literature (Duda et al., 2000) because it is known to be over conservative. The classification matrix obtained for the method is presented in the Table 3. The Kappa coefficient was $K=81.0\%$ with variance $1.6 \times 10^{-3} \%$. In 19 cases, the evaluation tool made mistakes and at least one classification was made incorrectly in all classes. That performance is good and shows that maximum likelihood method is a competitive approach in the solution of assessment problems.

Class of performance according to experts	Class of performance according to Evaluation Tool		
	1	2	3
1	46	0	4
2	1	49	0
3	8	6	36

Table 3. Classification Matrix for the Evaluation Tool based on Maximum Likelihood.

In statistical terms, the performance of maximum likelihood method is acceptable. About computational performance the average of CPU time consumed was 0.0160 seconds, showing the method is qualified as online.

6. Conclusion

The development of applications based on VR for medical training involves knowledge of several areas and demands a multidisciplinary team. The union of ideas is important in all stages of the development as: definition of scope, design, modelling, definition of assessment method, implementation, tests and validation.

A simulator based on VR must contain the same steps present in a real procedure and can use volumetric models, stereoscopic visualization, haptic systems, deformation methods and interactive cut. The use of all those features requires a lot of processing and can compromise the real-time performance. Usually, the development team defines the most important features to integrate the simulator. It will also depend on the budget available.

Several systems were developed to the design and implementation of VR applications. Some of them are not free and demands specific non-free platforms. There are also, free systems to allow the development, integration and synchronization of tasks and tools in VR systems. In particular, the framework CyberMed presents a higher level of abstraction, is dedicated to the development of medical applications and presents methods for on-line assess users' performance.

A bone marrow harvest simulator was developed with CyberMed, explaining their facilities when compared to the full implementation of techniques and devices support routines. Thus, this Chapter showed the challenges and possible solutions related to the development of VR-based simulators for training. Particularly, was presented the importance of a user assessment included in the system to allow identifying mistakes and points for improvement. In this context, CyberMed is the only framework that provides assessment methods ready for use.

In particular, an assessment tool can, in real-time, to modify simulation aspects, as variables and functions responsible for feedback. Besides, this information can also be utilized to verify the degree of accuracy of users when performing the simulation. In the bone marrow harvest simulator implementation was used an assessment tool based on maximum likelihood, which achieved good results in a performance test.

Nowadays, there are other simulators been developed with CyberMed. The inclusion of new classes and methods already allow the development of tutoring applications over Internet. However, the framework has been expanded with new functionalities for synchronous collaboration (Sales et al., 2010).

As future works, is intended to make tests and statistical comparisons among the others assessment methodologies presented in this Chapter, as well as, to make a statistical comparison between trainees that use the VR system in their training and other trainees that do not use.

7. References

- Burdea, G.; Patounakis, G.; Popescu, V.; Weiss, R.E. (1998) Virtual Reality Training for the Diagnosis of Prostate Cancer. Proc. IEEE Virtual Reality Annual Int. Symposium (VRAIS'98), pp. 190-197.
- Burdea, G & Coiffet, P. (2003) Virtual Reality Technology. Wiley Interscience.
- Cheng, J & Greiner, R. (2001) Learning Bayesian Belief Network Classifiers: Algorithms and System, Proc. 40th Canadian Conference on Artificial Intelligence.
- Correa, C.G.; Nunes, F.L.S.; Bezerra, A. (2009) Evaluation of VR Medical Training Applications under the Focus of Professionals of the Health Area. Proc. ACM-SAC Symposium on Applied Computer, pp. 821, Hawaii, ACM, Honolulu.
- Darzi, A.; Smith, S.; Taffinder, N. (1999) Assessing operative skills. British Medical Journal, v.318, n.7188, pp.887-888.
- Derossis, A.M. et al. (1998) Development of a model for training and evaluation of laparoscopic skills. American Journal of Surgery, v. 175, n.6, pp. 482-487.

- Dinsmore, M. (1996) Virtual reality simulation: training for palpation of subsurface tumors. Master's Thesis, Department of Mechanical and Aerospace Engineering, Rutgers University, October.
- Dinsmore, M.; Langrana, N.; Burdea, G.; Ladeji, J (1997) Virtual Reality Training Simulation for Palpation of Subsurface Tumors. Proc. IEEE VRAIS 1997, pp. 54-60.
- Dubois, D. & Prade, H. (1980) Fuzzy sets and systems - theory and applications. New York: Academic Press.
- Dubois, D. & Prade, H. (1988) Possibility theory: An approach to computerized processing of uncertainty. Plenum Press.
- Duda, R.O.; Hart, P.E.; Stork, D.G. (2000) Pattern Classification. 2nd ed. Wiley.
- Färber, M. et al. (2008) Heinz Handels Training and evaluation of lumbar punctures in a VR-environment using a 6DOF haptic device, Studies in Health Technology and Informatics v.132, p.112-114.
- Gallagher, A. G. et al. (1999) Virtual Reality training in laparoscopic surgery: A preliminary assessment of minimally invasive Surgical trainer Virtual Reality (MIST VR). Endoscopy, v.31, n.4, pp. 310-313.
- Gallagher, A. G. et al. (2001) Objective psychomotor skills assessment of experienced, junior, and novice laparoscopists with virtual reality. World Journal of Surgery, v.25, n.11, pp. 1478-1483.
- Gallagher, A. G. et al. (2005) Virtual Reality Simulation for the Operating Room: Proficiency-Based Training as a Paradigm Shift in Surgical Skills Training. Annals of Surgery, v.241, n.2, pp. 364-372.
- Huang, J; Payandeh, S.; Doris, P.; Hajshirmohammadi, I. (2005) Fuzzy Classification: Towards Evaluating Performance on a Surgical Simulator. Studies in Health Technology and Informatics v.111, pp. 194-200.
- Johnson, R.A. & Wichern, D.W. (2001) Applied Multivariate Statistical Analysis, Prentice Hall, 5 edition.
- Kumagai, K.; Yamashita, J; Morikawa, O.; Yokoyama, K. (2007) A New Force-Based Objective Assessment of Technical Skills in Endoscopic Sinus Surgery. Studies in Health Technology and Informatics v.125, pp. 235-237.
- Krause, P. J (1998) Learning Probabilistic Networks, Knowledge Engineering Review, vol. 13, pp. 321-351.
- Langrana, N.; Burdea, G.; Ladeji, J; Dinsmore, M. (1997) Human Performance Using Virtual Reality Tumor Palpation Simulation. Computer & Graphics, v.21, n.4, pp.451-458.
- Machado, L.S. (2003) Virtual Reality to Modeling and Simulation of Invasive Procedures for Pediatrics Oncology: A Case Study in Bone Marrow Transplant. PhD Thesis. Dept. of Electrical Engineering, University of Sao Paulo. (In Portuguese)
- Machado, L.S. & Moraes, R.M. (2006). Evaluation of Training Executed by Web Using Multilayer Perceptron Based Systems. Proceedings of World Congress on Computer Science, Engineering and Technology Education (WCCSETE'2006). March, Santos, Brasil, p. 228-232.
- Machado, L.S. & Moraes, R.M. (2009). Qualitative and Quantitative Assessment for a VR-Based Simulator. Studies in Health Technology and Informatics, v. 142, p. 168-173.
- Machado, L.S. & Moraes, R.M. (2010) Intelligent Decision Making in Training Based on Virtual Reality. In: Da Ruan. (Org.) Computational Intelligence in Complex Decision Systems, Chapter 4. Atlantis Press, 9789078677277, Paris.

- Machado, L. S.; Moraes, R. M.; Zuffo, M. K. (2000) A Fuzzy Rule-Based Evaluation for a Haptic and Stereo Simulator for Bone Marrow Harvest for Transplant. In: Annals of Phantom Users Group. Aspen.
- Machado, L. S.; Valdek, M. C. O.; Moraes, R. M. (2008) On-Line Assessment System for a Training System with Multiple Stages Based on Virtual Reality. *Journal of Multiple-Valued Logic and Soft Computing*, v.14, n.3-5, p.511-524.
- Machado, L.S.; Moraes, R.M.; Souza, D.F.L; Souza, L.C.; Cunha, I.L.L. (2009) A Framework for Development of Virtual Reality-Based Training Simulators. *Studies in Health Technology and Informatics*, v. 142, pp. 174-176. IOSPress.
- Mackel, T.; Rosen, J.; Pugh, C. (2006) Data Mining of the E-Pelvis Simulator Database: A Quest for a Generalized Algorithm for Objectively Assessing Medical Skill, *Studies in Health Technology and Informatics* 119, 355-360.
- Mamdani, E.H. & Assilan, S. (1975) An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, v. 7, no. 1, pp.1-13, 1975.
- McBeth, P.B. et al. (2002) Quantitative Methodology of Evaluating Surgeon Performance in Laparoscopic Surgery. *Studies in Health Technology and Informatics* v. 85. 2002, pp. 280-286.
- McCloy, R. & Stone, R. (2001) Science, medicine, and the future: Virtual reality in surgery. *British Medical Journal*, v.323, n.7318, p. 912-915.
- Moraes, R.M. & Machado, L.S. (2005a) Maximum Likelihood for On-line Evaluation of Training Based on Virtual Reality. *Proc. Global Congress on Engineering and Technology Education (GCETE'2005)*. March, Santos, Brasil, p.299-302.
- Moraes, R.M. & Machado, L.S. (2005b) Evaluation of Training Executed by Web Using Fuzzy Rule Based Expert Systems. *Proc. Global Congress on Engineering and Technology Education (GCETE'2005)*. March, Santos, Brasil, p.857-860.
- Moraes, R.M. & Machado, L.S. (2006) Continuous Evaluation in Training Systems Based on Web. *Proceedings of World Congress on Computer Science, Engineering and Technology Education (WCCSETE'2006)*. Março, Santos, Brasil, p. 223-227.
- Moraes, R.M. & Machado, L.S. (2007) Multiple Assessment for Multiple Users in Virtual Reality Training Environments. *Lecture Notes in Computer Science*. Berlin. v. 4756, p. 950-956. Springer.
- Moraes, R.M. & Machado, L.S. (2008) Using Embedded Systems to Improve Performance of Assessment in Virtual Reality Training Environments. *Proc. Int. Conference on Engineering and Technology Education 2008 (Intertech'2008)*. March, Santos, Brasil, p. 140-144.
- Moraes, R.M. & Machado, L.S. (2009a) Online Training Evaluation in Virtual Reality Simulators Using Possibilistic Networks. *Proc. Safety Health and Environmental World Congress 2009 (SHEWC'2009)*. July, Mongaguá, Brasil, p. 67-71.
- Moraes, R.M. & Machado, L.S. (2009b) Another Approach for Fuzzy Naive Bayes Applied on Online Training Assessment in Virtual Reality Simulators. *Proc. Safety Health and Environmental World Congress 2009 (SHEWC'2009)*. July, Mongaguá, Brazil, p. 62-66.
- Moraes, R.M. & Machado, L.S. (2010). Fuzzy Gaussian Naive Bayes Applied to Online Assessment in Virtual Reality Simulators. *Proc. 9th Int. FLINS Conf. Foundations and Applications of Computational Intelligence (FLINS'2010)*. August, Chengdu, China, p. 243-248.

- Moraes, R.M.; Machado, L.S.; Souza, L.C. (2009) Online Assessment of Training in Virtual Reality Simulators Based on General Bayesian Networks. In: Proc. VI International Conference on Engineering and Computer Education (ICECE 2009). Mongaguá, Brazil. CDROM.
- Neapolitan, R.E. (2003) Learning Bayesian Networks, Prentice Hall. Series in Artificial Intelligence.
- Rich, E. & Knight, K. (1993) Artificial Intelligence. McGraw Hill.
- Sales, B.R.A.; Machado, L.S.; Moraes, R.M. (2010) Interactive Collaboration for Virtual Reality Systems related to Medical Education and Training. In: Proc. 6th Int. Conference on Technology and Medical Sciences (TMSi), Porto, Portugal. CDROM.
- Sensable Technologies – Haptic Devices. Online: <http://www.sensable.com/products-haptic-devices.htm>. Access in July, 2010.
- Souza, D.F.L.; Cunha, I.L.L.; Souza, L.C.; Moraes, R.M.; Machado, L.S. (2007) Development of a VR Simulator for Medical Training Using Free Tools: A Case Study. In: Proc. of Symp. Virtual and Augmented Reality (SVR2007), p. 100-105, Brazil, SBC.
- Taffinder, N. et al. (1998) Validation of virtual reality to teach and assess psychomotor skills in laparoscopic surgery: results from randomised controlled studies using the MIST VR laparoscopic simulator. *Medicine Meets Virtual Reality*. Amsterdam: IOS Press, p. 124-130.
- Wilson, M.S.; et al. (1997) MIST VR: a virtual reality trainer for surgery assesses performance. *Annals of the Royal College of Surgeons of England*, v.79, p. 403-404.
- Zadeh, L.A. (1965) Fuzzy Sets. *Information Control*. n 8., pp. 338-353.
- Zadeh, L.A. (1988) Fuzzy Logic. *Computer*, v. 1. pp.83-93.

IntechOpen



Virtual Reality

Edited by Prof. Jae-Jin Kim

ISBN 978-953-307-518-1

Hard cover, 684 pages

Publisher InTech

Published online 08, December, 2010

Published in print edition December, 2010

Technological advancement in graphics and other human motion tracking hardware has promoted pushing "virtual reality" closer to "reality" and thus usage of virtual reality has been extended to various fields. The most typical fields for the application of virtual reality are medicine and engineering. The reviews in this book describe the latest virtual reality-related knowledge in these two fields such as: advanced human-computer interaction and virtual reality technologies, evaluation tools for cognition and behavior, medical and surgical treatment, neuroscience and neuro-rehabilitation, assistant tools for overcoming mental illnesses, educational and industrial uses. In addition, the considerations for virtual worlds in human society are discussed. This book will serve as a state-of-the-art resource for researchers who are interested in developing a beneficial technology for human society.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ronei M. Moraes and Liliane S. Machado (2010). Development of a Medical Training System with Integration of Users' Assessment, Virtual Reality, Prof. Jae-Jin Kim (Ed.), ISBN: 978-953-307-518-1, InTech, Available from: <http://www.intechopen.com/books/virtual-reality/development-of-a-medical-training-system-with-integration-of-users-assessment>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen