OPEN ACCESS
EURASIA Journal of Mathematics Science and Technology Education
ISSN 1305-8223 (online) 1305-8215 (print)
2017 13(2):395-416
DOI 10.12973/eurasia.2017.00623a

Effective Design of Educational Virtual Reality Applications for Medicine using Knowledge-Engineering Techniques

Filip Górski
Poznan University of Technology, Poznań, Poland
Paweł Buń
Poznan University of Technology, Poznań, Poland
Radosław Wichniarek
Poznan University of Technology, Poznań, Poland
Przemysław Zawadzki
Poznan University of Technology, Poznań, Poland
Adam Hamrol
Poznan University of Technology, Poznań, Poland

Received 21 March 2016▪ Revised 13 May 2016▪ Accepted 15 May 2016

ABSTRACT

Effective medical and biomedical engineering education is an important problem. Traditional methods are difficult and costly. That is why Virtual Reality is often used for that purpose. Educational medical VR is a well-developed IT field, with many available hardware and software solutions. Current solutions are prepared without methodological approach. As a result, each VR system used for medical purposes is unique - there is no standardization, which leads to increased development time and costs. The paper presents an original methodology that can be used for building immersive virtual reality (VR) applications dedicated to training medical skills. The methodology is named "knowledge-oriented medical virtual reality" (KOMVR), as it uses basic knowledge-engineering tools during processes of planning, building and verification of interactive educational solutions. The paper shows successive stages of the methodology and the tools used for its execution. Effectiveness of the proposed methodology is illustrated with a simple example – a human body atlas. Major time reduction can be achieved while using the methodology - time reduction in the presented case was from 50 to 14 working days. The proposed innovative approach can increase effectiveness of preparation of educational VR solutions in medicine, but also in other branches of engineering.

Keywords: virtual reality, knowledge engineering, Medical education, computer-aided teaching methods

Correspondence: Filip Górski, *Poznan University of Technology, pl. M.Skłodowskiej-Curie 5, 60-965 Poznań, Poland.*
filip.gorski@put.poznan.pl

State of the literature

- Educational medical VR is a well-developed information technology field, with many different dedicated hardware and software solutions.
- Presently, there is high demand for such solutions, as teaching medical procedures is difficult and costly, whereas using virtual, digital models can reduce risk and costs, and provide a realistic experience that maintains the educational effect.
- Current solutions in this field are usually prepared without any methodological approach. As a result, each VR system used for medical purposes is unique both in hardware and software.

Contribution of this paper to the literature

- This paper proposes to use well-known knowledge-engineering tools, namely MOKA (methodology and tools oriented to knowledge-based engineering applications) as a basis of an entirely new methodology (KOMVR) of conduct while building educational medical VR applications.
- A new method of classification of VR applications is proposed, as well as a new approach to planning, design and knowledge acquisition processes, based on ICARE forms (Illustration, Constraint, Activities, Rules, Entities) from MOKA.
- This innovative approach can increase effectiveness of preparation of these types of applications, which is illustrated by a case study.

INTRODUCTION

Immersive virtual reality (VR) has been known for decades, but has gained considerable recognition and has found a wide range of professional applications in the beginning of this century. The main concept of VR is creation of a digital world (or environment), in which a human user is placed and can interact with it in real time (Burdea & Coiffet 2003). Virtual reality is distinguished from other human-computer interaction methods by 3D graphics striving at realism, intuitive interaction and user immersion inside the digital scene (Burdea & Coiffet 2003, Scherman & Craig 2003). Immersion is a feeling of being present inside an artificial environment, despite physical presence in the real world (Bowman 2007), and is achieved mostly by use of various stereoscopic projection devices, such as head-mounted displays (HMD) or CAVE systems (Cruz-Neira et al. 1993), joined with user tracking solutions for enhancement of feeling of presence inside the virtual world. In recent years, new low-cost devices have emerged on the market (e.g. Oculus Rift 2014). Computing power is also steadily rising, allowing more demanding real-time visualizations and simulations. Also, access to cheap or free software for building interactive virtual worlds is continually increasing. Therefore, it can be said that building VR solutions is becoming easier and easier.

A VR environment can heavily influence behavior and feelings of a user and can be used for teaching knowledge and skills. The positive educational effect of VR has been observed and reported in literature as early as the 90's (Bell & Foegler 1995) and proven in further studies (Getchell et al. 2010). Today, VR is commonly used in engineering education (Abulrub et al. 2011, Grajewski et al. 2015), as well as in many other engineering activities, including simulations of machine operation and assembly (Pandilov et al. 2015, Seth et al. 2011), knowledge visualization (van Dam et al. 2000), and configuration of variant products in the automotive branch (Górski et al. 2015, Jiang 2011). Virtual reality has many medical applications (Riva 2003) and is often used for educational purposes (Stansfield et al. 2000). There are many different possibilities of use of VR in medical education, from simple applications that can replace a book, such as virtual anatomical atlases (Hamrol et al. 2015), to more complex presentations, such as applications for full, realistic simulation of surgical procedures (Zhaoliang 2011, Vankipuram et al. 2014). For realistic simulation, haptic devices with force simulation are frequently used, as it allows trainees to develop required abilities before first contact with a real patient (Escobar-Castillejos et al. 2016). Low-cost VR devices can also be used effectively in medical education (Glatter 2015, Buń et al. 2015).

Virtual reality in medical education has a number of advantages – mainly lower costs (as opposed to real models, tissues or patients) and less risk and stress for students. Despite all the advantages, it is still not commonly used in medical universities worldwide. The first reason for this is that hardware is costly, although this is presently changing thanks to cheap VR solutions such as HTC Vive or Oculus Rift (Dingman 2015). The second and most important reason is that building a VR application for medical education purposes requires, on one hand, high programming qualifications and, on the other hand, lots of specialized medical knowledge. Development of effective VR learning tools for medical students requires time and investment, as well as full cooperation between doctors and software developers.

In the opinion of the authors', the main hindrance to VR being commonly applied is that most solutions are immediate, created just for the purpose defined in current context, without looking back at any methodology that could make the building process faster and obtain more effective results. There is no known dedicated methodology of planning, building and implementing educational VR solutions in medicine. As each solution is immediate, the time and costs are high, with a frequent number of mistakes and corrections.

On the basis of their own experiences in building education VR medical solutions, the authors of this paper propose a new methodological approach. Using experience in building of engineering-aiding solutions based on knowledge processing, known commonly as knowledge-based engineering (KBE) (Reddy et al. 2015), a set of general guidelines about a process of planning, building and implementing educational medical VR applications is presented. In the KBE systems, identified and gathered expert knowledge about what, when and how it needs to be done is processed by computer systems, allowing its easier application in new projects. A formal description of rules used by the design engineers influences the standardization of the process and allows shortening it (Verhagen et al. 2012). The KBE systems are commonly used for design and process automation (Zawadzki et al. 2011, Reddy et al. 2015, Tiwari et al. 2013) and there are many available, positively verified methodologies. The main idea of this paper is use of some available knowledge-engineering tools, such as MOKA (Methodology and tools Oriented to Knowledge-based engineering Applications)

(Oldham et al. 1998), to improve and standardize the process of building educational medical VR applications. The whole process is demonstrated in further sections and proven effective by presented case studies.

MATERIALS AND METHODS

Virtual reality systems

A VR application is a software application (a computer program designed to perform a group of coordinated functions, tasks, or activities for the benefit of the user) that usually contains high-quality 3D graphics. To be considered true VR, the application must use special, dedicated hardware, for both projection and interaction, aimed at achieving the feeling of immersion. Software together with hardware forms a VR system (**Figure 1**). Components of the system are divided into software, such as a VR application usually created on the basis of externally prepared 3D and 2D data, using a specific engine and programming methods, and hardware, for projection and intuitive interaction (**Figure 2**).

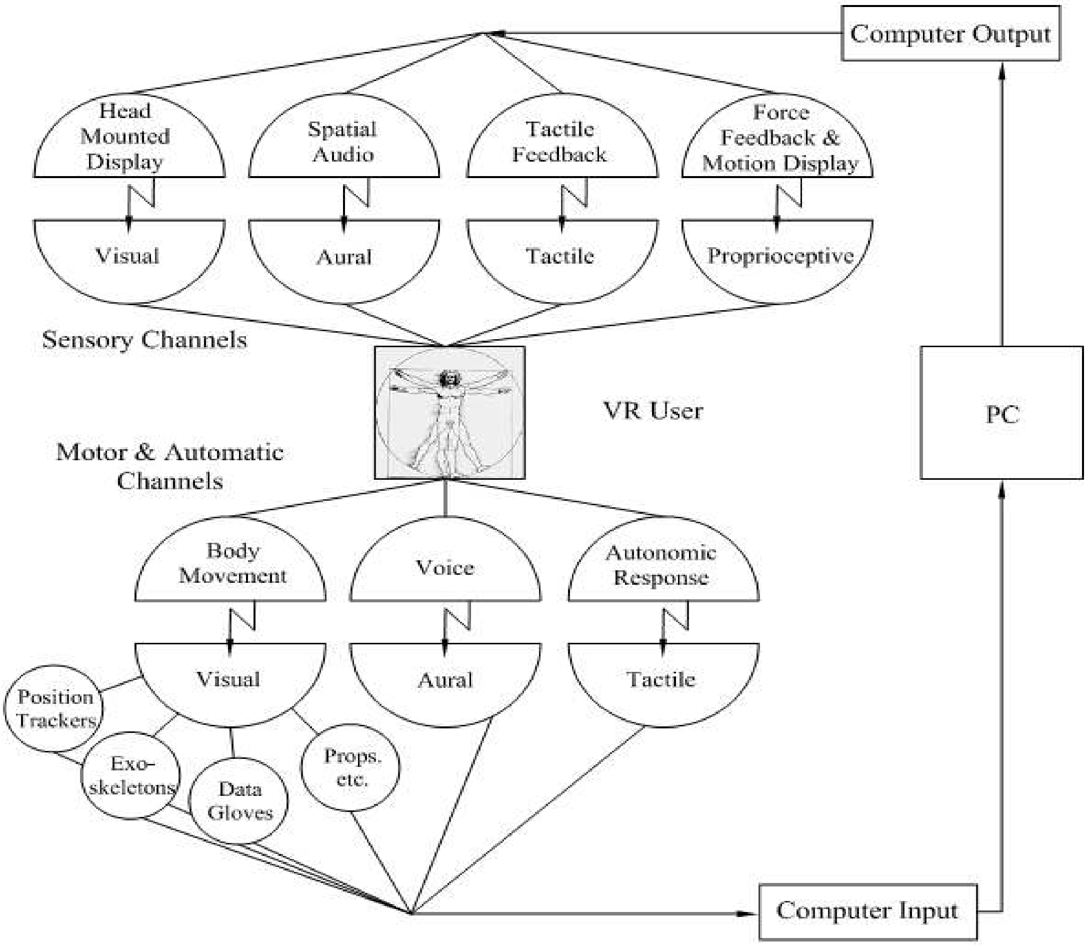

Figure 1. Virtual Reality system (Biocca & Delaney 1995)

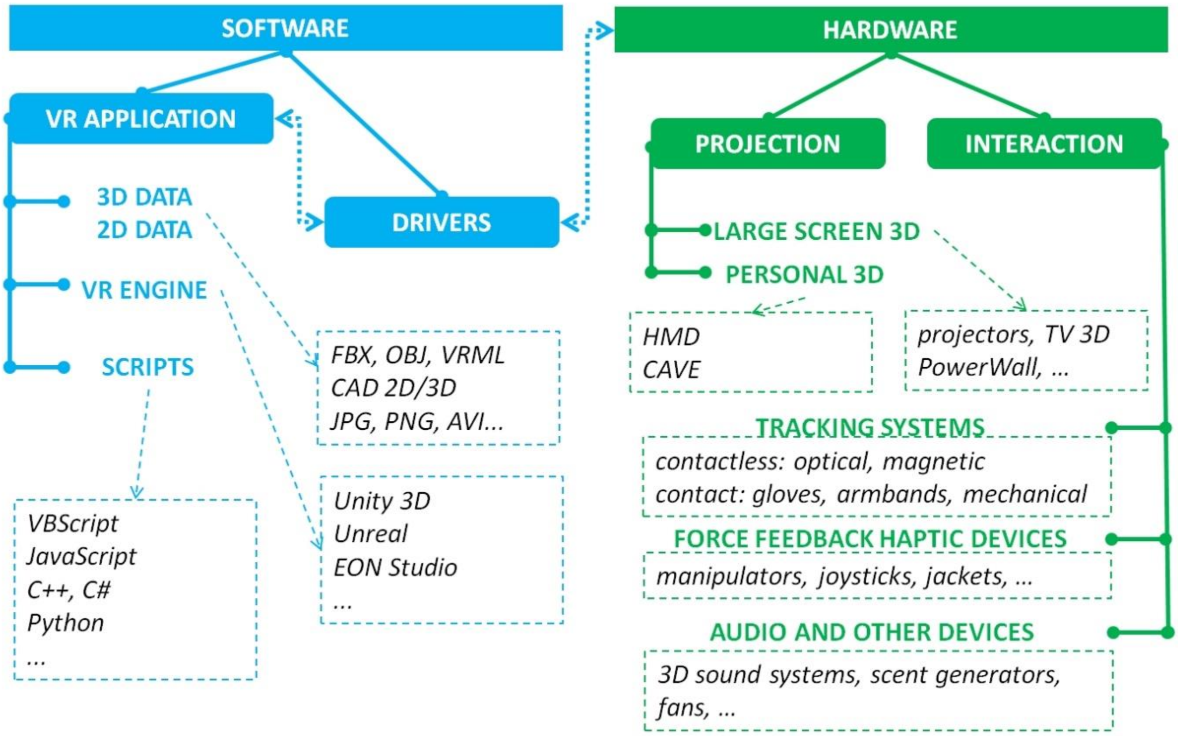

Figure 2. VR system components

There are countless possibilities of composing VR systems. There are numerous hardware components, for instance for interaction, with better or worse capabilities, such as accuracy, working area, degrees of freedom, latency, etc. (Gutierrez et al. 2008). There are no standardized procedures or ready packages, which means after analysis of just a few available applications, a conclusion can be drawn that almost every VR system is unique and built from scratch based on available components (De Mauro 2011, Greuter & Roberts 2015, Riva 2003). This is not a favorable situation, as it can lead to use of improper components and can result in a system without full effectiveness, especially because there are practically no studies available dedicated to comparison between different systems and their efficiency in various applications, particularly in medicine. One of the aims of this paper is to present a set of basic guidelines on how to perform a preliminary selection of classes of components and what tools to select to make a final decision

Knowledge-engineering methodologies

Knowledge engineering is a discipline that deals with problems of acquisition, representation and application of knowledge in computer systems. It is a discipline related to creation of knowledge bases and use of semantic technologies to process knowledge by computer systems (Schreiber 2007). It is comprised of:

- identification of knowledge sources
  - acquisition (obtaining) of knowledge

- representation of knowledge
- analysis of identified knowledge
- creation of bases and repositories of knowledge
- searching, accessing and sharing knowledge

Efficient use of knowledge in the design process requires its appropriate acquisition and proper recording in a knowledge base. Methods of knowledge gathering are also dependent on the sources, which can be of variable quantity and form. The most important knowledge sources are:

- personal notes
- technical documentation (drawings, schemes)
- information from previous projects
- study results (e.g. from simulation or operation studies)
- libraries of documents, books, papers
- standards and catalogs
- remarks from research centers and patent offices
- remarks from customers and suppliers

Representation of knowledge is understood as its formal recording. The method of representation should be as simple, complete, comprehensible and unequivocal as possible, not only for persons dealing with its description (i.e. knowledge engineers) but most of all for those who will use it. For creation of ontology (i.e. description of concepts and relations characteristic for a given knowledge branch), specific notations and languages are used (e.g. Unified Modeling Language).

A basic problem while building knowledge-based systems is the problem of ordering the knowledge. Realization of tasks related to this problem, because of their similarity in various disciplines, contexts and applications, was attempted to be simplified by development of various methodologies aimed at supporting the building of systems based on knowledge. Over the past two decades, many solutions have appeared, aiding the development of knowledge-based systems of general purpose, such as the CommonKADS methodology (Common Knowledge Acquisition and Documentation Structuring) (Schreiber 2007). On the basis of CommonKADS, MOKA was created and is now commonly used in engineering systems (Stokes 2001). MOKA, being a result of interdisciplinary work of scientific teams and representatives of aeronautical and automotive industry, was prepared to achieve the following main goals (Oldham et al. 1998):

- reduction of risk, time and cost of development of KBE applications by 20-25%
- ensuring development and maintenance of KBE applications
- preparation of tools supporting application of the method
- introduction of international KBE standard

The creators of MOKA divided the work of building a KBE system into the following stages:

- identification – determination of purpose and range of building of the KBE system
- justification – approximation of resources, costs and business risk
- acquisition – gathering the knowledge from selected sources
- formalization – formal recording of the gathered knowledge
- application – implementation of knowledge in computer software
- implementation – launching of the KBE system

In the beginning, the aim of building the system needs to be determined, available resources must be estimated, a plan of the project should be prepared and the need of its building must be justified. Next, out of the indicated sources, the knowledge must be collected and recorded in a way to ensure possibility of its implementation in the chosen computer software. Stages related to identification and justification apply to organizational and economical aspects of building a KBE system, and in reality are omitted in MOKA, as are the two final stages – application and implementation – which concern development of an application, its installation, use and testing. MOKA focuses on stages of acquisition and formalization, which apply to methods of collecting and converting knowledge into language understandable for a computer application.

METHODOLOGY OF BUILDING MEDICAL VR APPLICATIONS USING KNOWLEDGE ENGINEERING

General concepts

The authors of this paper, on the basis of MOKA and their own experience in creating VR solutions, propose an original methodology of building educational medical VR applications. This methodology was named "knowledge-oriented medical virtual reality" (KOMVR) and this name will be used henceforth.

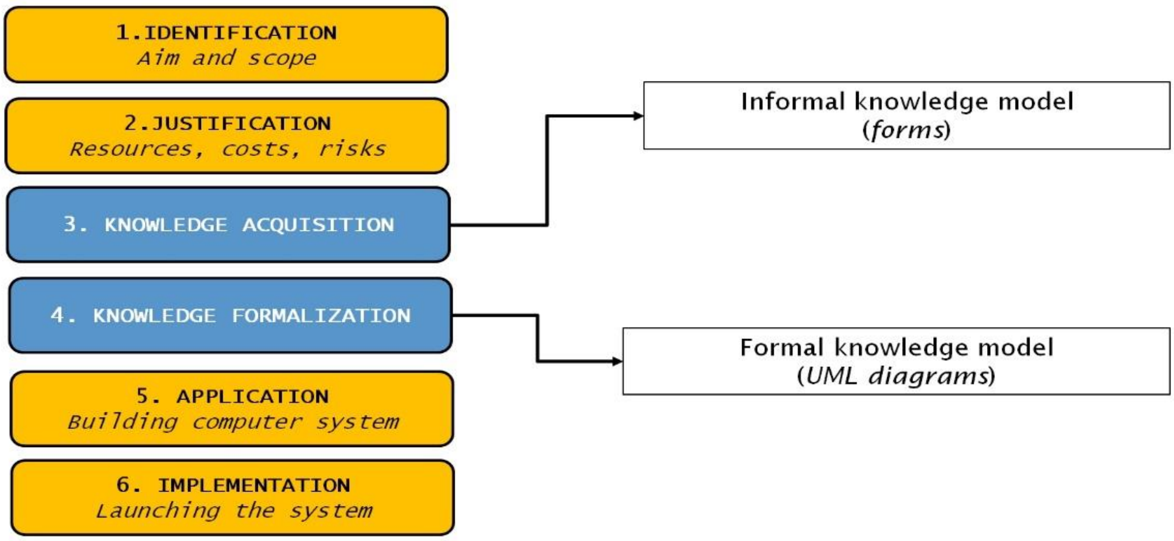

Figure 3. Basic stages of building a VR system in KOMVR based on MOKA – authors' own work based on (Oldham et al. 1998)

The basic stages distinguished by MOKA remain unchanged in the authors' proposed methodology (**Figure 3**). KOMVR is distinguished by the following features, added to the original MOKA approach:

- classification of educational VR applications into three different levels of contained knowledge, as well as for three different groups of end users
- indication of medical knowledge sources and necessity of their use at different stages of VR system development on the basis of classification (guidelines for composing an interdisciplinary team of specialists and its composition)
- guidelines for selection of VR hardware components (general level) based on classification of designed application
- recommendation of a set of tools for selection of VR software and hardware (specific solutions) at the "application" (fifth) stage of system development
- guidelines for verification of the created system at the last stage (implementation)

The aim of this methodology is to accelerate preparation of VR solutions (VR systems) for medical education by presenting a set of general procedures, which can be used for many different cases. Further sections of this paper provide more information about specific stages of the proposed methodology.

Identification and justification

Every VR application starts with definition of need (expectations) and purpose. At the conceptual stage, two basic questions need to be answered:

- Who will be using the application? Possible answers: medical students, young doctors, experienced doctors.
- What is the purpose of the application? Possible answers: obtain new basic knowledge, obtain specialized knowledge and/or basic skills, obtain new knowledge and advanced practical skills.

On the basis of the answers to the questions above, an application can be further classified to one of three levels of knowledge presentation, as shown in **Figure 4**. Classification of an educational VR application can be performed using a diagram with three distinct areas (**Figure 5**), where an answer to the two above-mentioned questions can be marked as a point and area in which the location of the point corresponds with a specific level of knowledge, and therefore a type of VR educational application. The diagram shows a general idea based on the authors' experience from both literature study and building educational medical VR applications (in cooperation with doctors of medicine), and should be treated as a basic guide to decide which level of VR application is built.

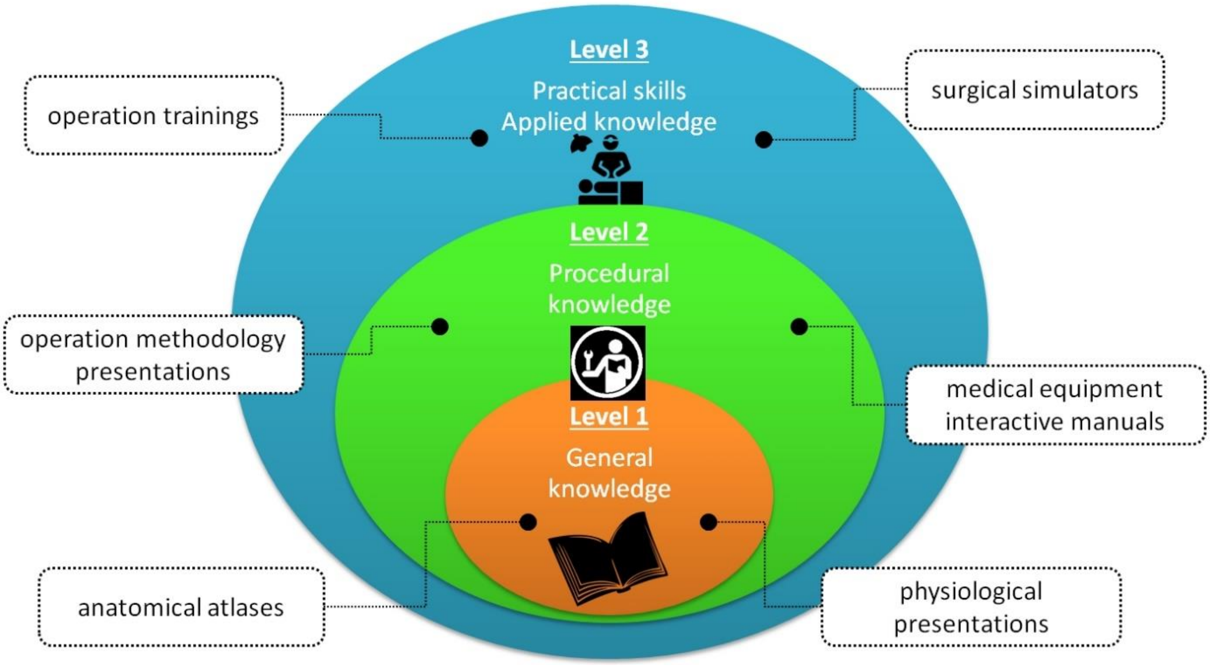

Figure 4. Levels of educational medical VR applications

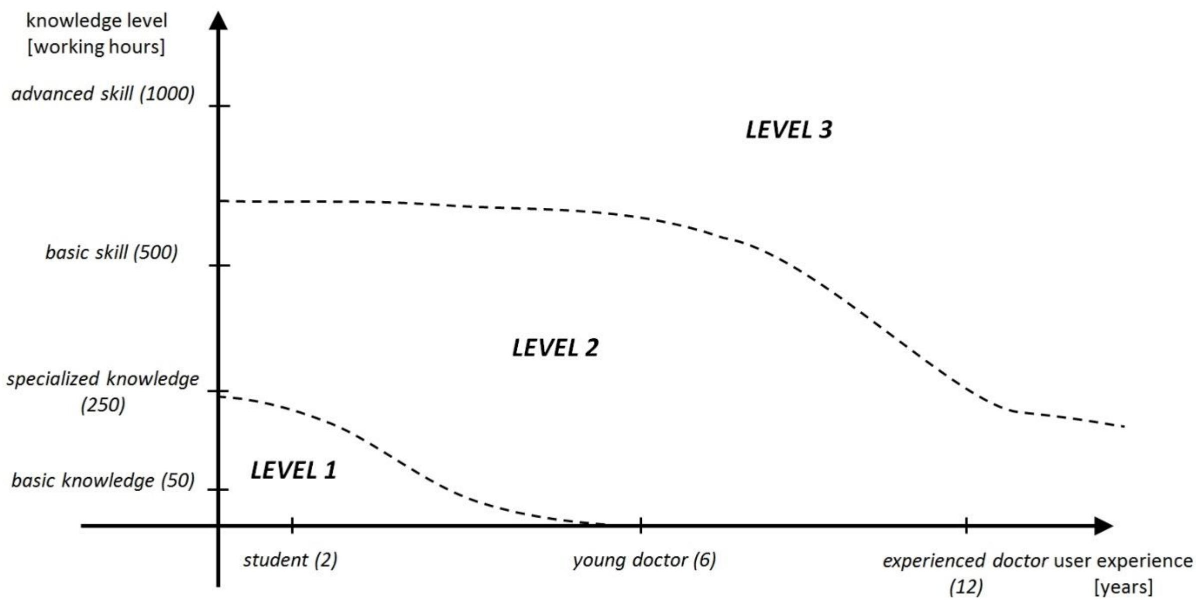

Figure 5. Diagram for selection of knowledge level of medical VR application

Each level of applications has its own set of features and requirements, such as necessary types of data, means of interaction and expected behavior of presented virtual objects. These features correspond with specific software and hardware components of a VR system. Possible features and their link to hardware and software capabilities necessary in building and implementation of a VR system are presented in **Table 1**. It is noteworthy that subsequent levels contain the lower levels – an application from Level 2 should have more or less the same contents in terms of complexity as Level 1, expanded with more specialized data, interaction, etc.

Table 1. Features and requirements of applications at different knowledge levels

Features/requirements	Level 1	Level 2	Level 3
Visualization	Static – pre-rendering (possible static images and pre-rendered sequences)	Kinematic – real-time rendering (3D engine, live animation)	Dynamic – real-time rendering with object deformations
Human tissue data form	Illustrative (3D modeled by graphic designer with no medical imaging)	Selected cases (modeled on the basis of pre-selected patient data from medical imaging)	Real data (data from CT scans or other medical imaging, processed for better visualization)
Animations	Simple, pre-rendered	Rigid bodies – real-time, deformations – pre-rendered	Both rigid and deformable bodies in real time
Object manipulation and interaction methods	Mouse, keyboard, graphical user interface, gestures	Graphical user interface, gestures, tracking	Graphical user interface, gestures, tracking, haptic manipulation with force feedback
Collisions and force feedback	Unnecessary	Beneficial	Mandatory if no physical props*
Full immersion (HMD)	Beneficial	Needed but not mandatory	Mandatory if no physical props*
Required tracking and force accuracy	Low or N/A	Medium/low	High
Required computing power	Low/medium	Medium/high	High
Participation of specialists – medical doctors	Planning stage – definition of requirements	Planning and building stage – sharing procedural knowledge	Planning, building and verification – recording of procedure

* force feedback and full immersion are not required when there are hardware components representing the doctor's workplace, such as physical patient phantoms, physical tools, etc.

When applied to a specific case, **Table 1** can be used to determine what classes of components should be included in the VR system. For instance, an application for teaching a procedure of limb amputation, belonging to Level 3, requires a haptic device, an HMD and a 3D engine with support of real-time geometrical deformation and collisions. The determined features can also be used to find out what kinds of specialists are required in a team that is building the application.

After determining the type of application and its basic features and requirements, sources of knowledge must be defined. For different knowledge levels, different sources should be selected, as presented in **Table 2.**

Table 2. Knowledge sources for an educational medical VR application

Level	Possible sources	Types of knowledge
1 – general knowledge	• books • photographs • lecture materials	• descriptions • illustrations • technical data
2 – procedural knowledge	all the above-mentioned, plus: • medical procedure videos • notes and guidelines from specialists	• procedure illustrations and schematics • technical documentation • selected cases
3 – practical skills	all the above-mentioned, plus: • specialist muscle memory • human body measurement data (CT)	• recorded movements of specialists • 3D measurements

After identifying knowledge sources and selecting basic components of the VR system, a risk analysis must be performed. It should be performed in the following order:

1) Planning composition of an interdisciplinary team for building and implementing the planned application. The team should consist of VR specialists (e.g. programmers, graphic designers), as well as of medical specialists (e.g. surgeons, technicians, doctors). The possibility of assembling such a team should be assessed and considered in the risk analysis.

2) Calculation of working hours. For each member of the team, appropriate tasks must be defined. The tasks start from knowledge gathering (see following sections), through its formalization and implementation in the final application. Time must be calculated along with costs and considered in the risk analysis.

3) Calculation of hardware and software costs. Basic assumptions should be made about price limits for particular classes of components, to limit their selection to particular groups. The final price should be considered as another risk factor.

4) Estimation of market size and financing – this is a wide problem. In general, if the solution will be commercially spread, the market must be determined and a business plan must be prepared. In the case of an internal solution (e.g. inside a medical university or company), sources of funding must be found.

5) Summary. Taking into account all the above-mentioned risk factors, final risk should be calculated and a decision should be made whether to continue the project or not.

Knowledge acquisition and formalization

Knowledge acquisition and formalization are the first tasks in the actual building process of educational medical VR application. At the knowledge acquisition stage, the necessary knowledge about the problem and the planned application should be gathered and informally recorded in forms. MOKA makes precise assumptions about this stage – the so-called ICARE forms should be used. The acronym ICARE stands for:

- Illustration – used for representation of all types of general knowledge – figures, descriptions, comments
- Constraints – used for modeling relations between entities
- Activities – used for description of various actions in the realized process
- Rules – used for description of rules related with activities in the realized process
- Entity – used for description of object class (structure, functions, behaviors)

In the presented KOMVR, the same ICARE forms can be used, although with a different approach. The authors suggest the following order of operation while gathering knowledge for medical VR applications:

1) Entities – descriptions of structural and functional entities in the VR application. The structural entities will be 3D geometry – of human tissues, medical equipment, etc., as well as 2D components, such as elements of a graphical user interface. The functional entities will be linked to 3D geometry – they can be actors (objects manipulated by the user directly, such as surgical tools in Level 3 apps), elements of environment (e.g. an operating room and its equipment) or interactive objects (e.g. tissues, organs of virtual patient).

2) Constraints – defined constraints between entities allow description of behavior of objects. The constraints in VR applications can be used mostly for defining object hierarchy (which objects belong to common groups) and connections (type and character of link between 3D objects).

3) Activities – description of actions that can be realized in the VR applications, i.e., operations that can be performed by the user directly. They can be related to user movement (using tracking systems) or other actions such as selecting options from a graphical menu, causing specific animations or other behaviors of objects. The activities are composed of sensors (triggers) and behaviors – a specific signal from a sensor triggers a specific behavior. The path between sensors and behaviors can be determined by use of rules.

4) Rules – determine if a specific action can occur when a sensor is triggered. For example, an artificial heart valve cannot be inserted if the biological valve is not removed, even if the user makes an appropriate action.

5) Illustrations – figures, descriptions and diagrams defined for each entity, constraint, activity and rule, allowing clear understanding of how and why it works. For example, in a 3D anatomical atlas, illustrations of entities – particular organs and

tissues – can be represented by figures (photographs and drawings) with text descriptions. Illustration of rules and constraints can be presented in the form of graphs or a series of figures. A general rule proposed by the authors is that each distinguished object should have its own illustration. Exemplary illustrations of entities and activities are presented in **Figure 6.**

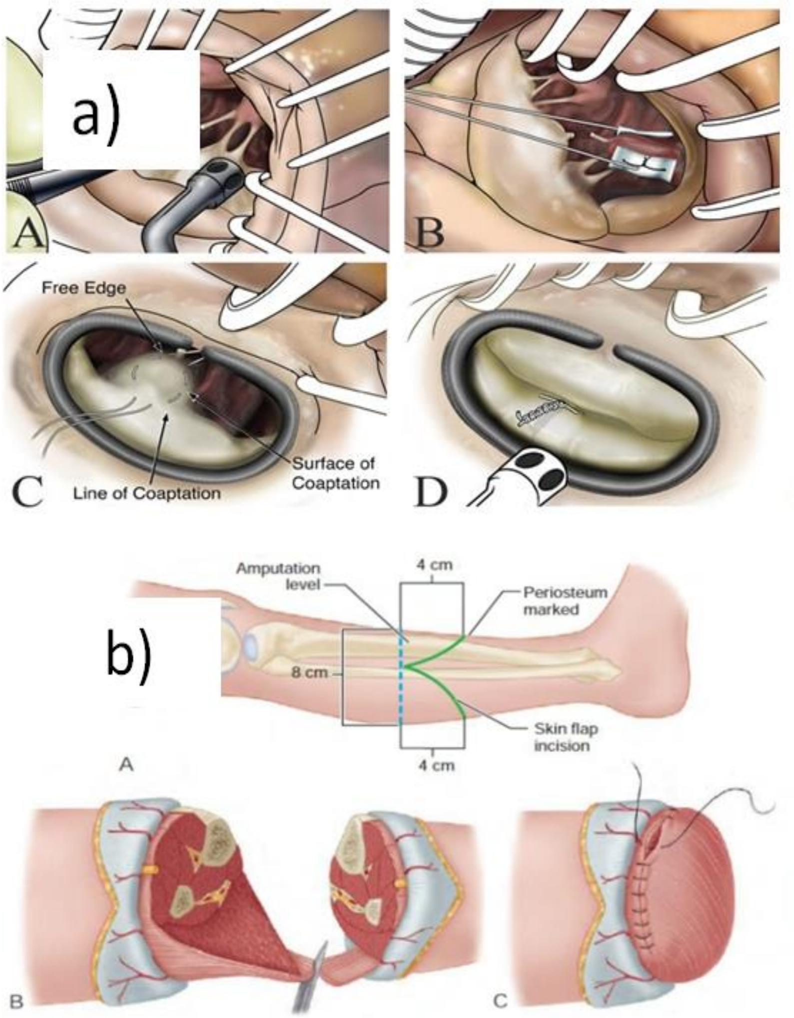

Figure 6. Exemplary illustrations of entities and activities with rules for mitral valve repair (a) and lower limb amputation (b)

The informal knowledge model as recorded in ICARE forms must be formalized at the next stage. The formalization in KOMVR is understood as recording the knowledge in a form that can be directly implemented in the VR system. The generally acceptable forms are diagrams in Unified Modeling Language, or other types of block diagrams, which must unequivocally determine how the virtual objects should behave after certain actions are taken by the user.

After knowledge formalization, participation of medical specialists is no longer needed, regardless of the level of application, as the formalized knowledge has a form understandable by VR programmers and can be directly implemented using different methods of programming. The method of programming depends on the selected VR engine, as some engines use classic object-oriented programming and/or scripting, while some use visual programming methods, where Unified Modeling Language diagrams can be directly transferred.

Application and implementation

The stages of development of the VR medical application in KOMVR, after knowledge is gathered and formalized, should be as follows:

1) Selection and obtaining of specific software and hardware components (planning).
2) Preparing non-interactive visualization with full navigation (visualization).
3) Programming interaction and behaviors of objects (programming).
4) Building a user interface and ensuring final user communication methods (user interface).
5) Internal verification and publication for use by a sample of end users (verification).

The activities performed at specific stages are presented in **Table 3**. Tools for these activities are also mentioned. The stages are similar to stages that can be found in the process of building any VR application. Detailed guidelines of building realistic VR applications are far beyond the scope of this paper and have been a subject of numerous research (Burdea & Coiffet 2003, Gutierrez et al. 2008, Scherman & Craig 2003), so only basic concepts will be described here with new input by the authors.

Table 3. Summary of stages of VR medical application building

Stage	Activities	Tools	End result
1 – planning	Defining criteria of selection, using analytical methods for choosing components	Decision-aiding tools: Analytical Hierarchy Process, fuzzy logic, cluster analysis	Selected and obtained software and hardware – BOM of VR system
2 – visualization	Preparation of 3D and 2D models, export-import procedures, hierarchy and navigation	3D modeling tools, 2D graphics tools, VR software engine	Non-interactive 3D visualization with navigation
3 – programming	Planning actions on objects, triggered by specific sensors	VR software engine, programming software*	Interactive visualization
4 – user interface	Application of graphical user interface, connection of all hardware components	VR software engine, programming software*	Complete VR application
5 – verification	Internal tests with or without medical specialists	VR software engine, survey studies	Guidelines for application improvement

* Optionally, if needed for development of additional, external modules

The planning stage should consist of selection of specific hardware and software components. A set of questions must be answered to achieve this. In general, it should consist of defining a matrix of criteria and alternatives for each possible component of the VR system (the general composition of the system is determined during the identification and justification stage) and then selection of the best possible variant. If this is not possible to do directly, for instance because of lack of distinct, comparable, quantitative criteria, there are methods that can help in making a decision. Two examples of methods that can successfully be used to aid a multi-criterial decision problem are the Analytical Hierarchy Process and the Technique for Order of Preference by Similarity to Ideal Solution. Regardless of the selected method, a final result of the first stage should be presented in a form of "bill of material" of the VR system, meaning a list of components necessary for building and implementing the prepared solution.

The visualization stage consists of preparing 3D models and importing them to a VR engine selected in the previous stage. At this stage, a number of visualization techniques should be applied, such as assignment of various materials, preparation and mapping of textures, normal maps or reflection maps to improve appearance of objects. Also, a number of effects can be applied in the VR engine to make 3D data look more realistic, for instance real-time shadows, ambient occlusion or other lighting effects. This stage can be performed almost entirely in a separate software for 3D modelling and visualization, as preparation of 3D models and mapping textures is not possible in most VR engines. The final task at this stage is preparation of hierarchy of objects to clearly represent the contents of the application and relations between entities recorded in the knowledge acquisition phase. An example of a final result of this stage is shown in **Figure 7**.

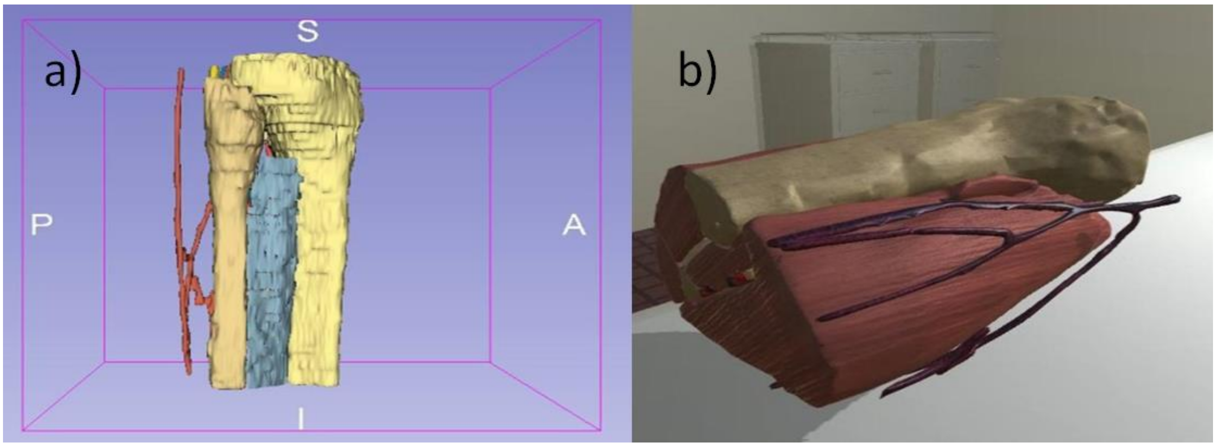

Figure 7. (a) Raw 3D data (b) data after processing, import and application of visualization techniques

The third and the fourth stage of preparation of the VR application are the most labor- and time-consuming activities of the whole process. The two phases can be considered separately, but are often conducted at the same time. The general goal here is to obtain a fully interactive application with all assumed functionalities regarding object manipulation, animation, interaction by means of both VR devices and classic interface (e.g. a graphical user interface and typical peripheral device, such as a keyboard and mouse), as well as logical integrity. There are many ways of programming the VR systems, starting from simple visual programming, to scripting, to object programming in high-level languages. Different VR engines (e.g. Unity 3D, EON Studio, Vizard, Unreal Engine) allow different methods of programming. In general, the authors propose description of the process of programming by four distinct classes of objects and relations between them. The objects are (see Figure 8):

- sensors – input points, where the signal comes in directly from the user or indirectly, for example as a result of collision caused by a user's actions; the sensors can be hardware (e.g. a signal from peripheral devices, tracking systems, haptic devices) or software (e.g. collisions, timers)
- inhibitors – used for blocking the signal between sensors and actions under certain conditions; a sensor should not always trigger an action and inhibitors are various logical conditions or timers that prevent them from doing so
- actions – all types of possible object behaviors, mostly classified either as animations (changes of position and orientation of objects or their parts over time) or as appearance changes (changes of properties such as color, texture, scale or opacity over time)
- components – 3D or 2D objects, or logical components that form the visualization, to which the actions are assigned by means of hierarchy

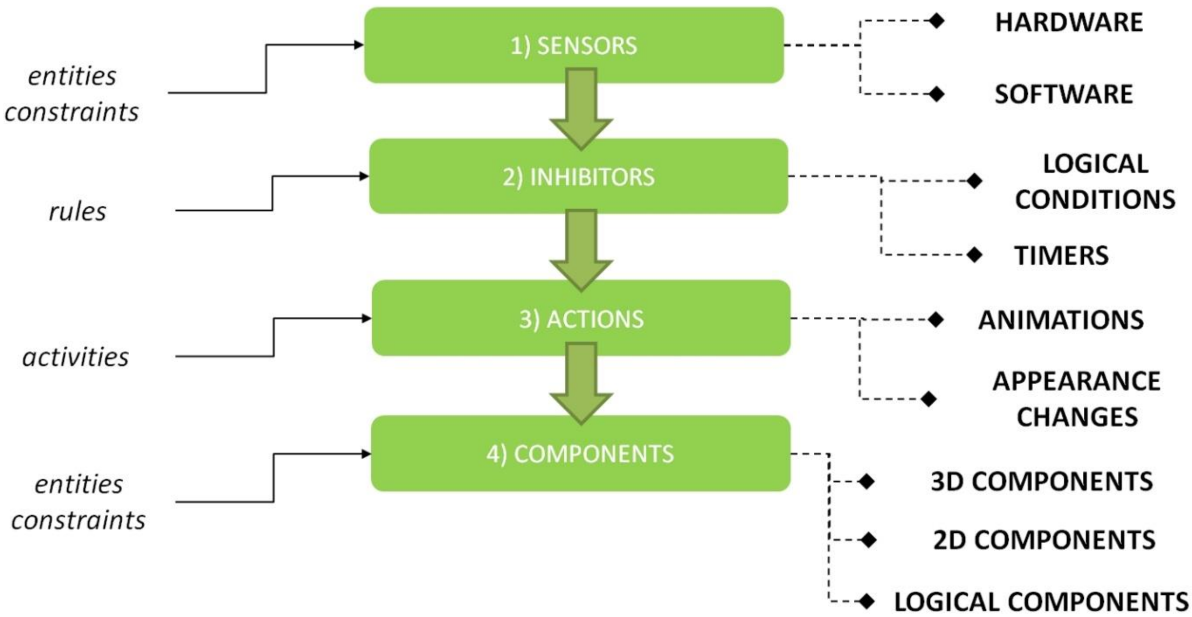

Figure 8. Programming of interaction in VR systems – basic object classes

These four object classes in the proposed KOMVR are derived directly from ICARE forms filled in at the knowledge acquisition stage. The entities and constraints are represented by components, sensors and their hierarchy. The activities translate into actions, while inhibitors are representatives of rules. The representations are not always 1:1, as one rule or constraint can be represented by multiple inhibitors and sensors. There could also be components, actions and sensors not distinguished in the ICARE forms.

An example of this approach could be a beating heart animation in a heart transplantation application. The heart is a component (an entity) here. Its basic activity is beating, represented by an animation (an action). The animation is triggered by a basic time sensor, starting from the beginning of the VR simulation (the simulation and its running time is an example of an entity bound with a constraint). The animation must be stopped when the heart is removed from the virtual patient's chest, so the action of heart removal (another animation, related with other entities, such as surgical tools) will start an inhibitor of the heart beating. The heart is connected to the body (a constraint) and its animation must be stopped when this connection is broken (a rule).

The result of the programming and user interface creation stage is a working, interactive application, ready for verification. The verification stage should consist of two phases – internal verification, with participation of the whole development team, and external verification, where the first version of the application should be evaluated by a sample of end users (in the case of educational medical VR applications, medical students or doctors, as well as patients). Participation of medical specialists at the internal verification stage is determined by the level of the application – see **Table 1** for guidelines. At Level 3 (teaching practical skills in medicine), participation and approval of medical doctors – specialists in a given field – is mandatory, according to KOMVR. The verification stage should result in an indication of a list of critical and non-critical corrections to be made in a final application. After they are introduced, which is usually an iterative process, the whole VR system (consisting of dedicated, selected hardware and the software application) can go into the distribution stage. If it is a single solution (a frequent case in Level 3 applications), made internally, for example, at a medical university, there is no distribution but rather an instant implementation. If the solution is intended to be distributed widely as a commercial product, it must undergo a commercialization procedure.

CASE STUDIES

The virtual 3D human body atlas

The virtual human body atlas application was made with participation of the authors, as a basic tool for learning anatomy for students of a medical university and biomedical engineering studies at a technical university. The purpose of the application is to effectively teach some general knowledge about human anatomy to first- or second-year students. According to the diagram in **Figure 1**, this allows classification of the application into Level 1. The following detailed features were distinguished in the application:

- selective, static visualization of all organ groups (e.g. skin, muscular system, bone system), hiding and showing all groups separately
- semi-opacity mode for selected or all organ groups

- section mode for dynamic sectioning of selected or all organs
- exploding view, the possibility to see all organ groups simultaneously, but separately (**Figure 9**)
- text or 2D graphical information about selected organ groups and body parts
- stereoscopic projection on a low-cost HMD
- interaction using low-cost tracking systems and gesture recognition devices, as well as standard peripheral devices

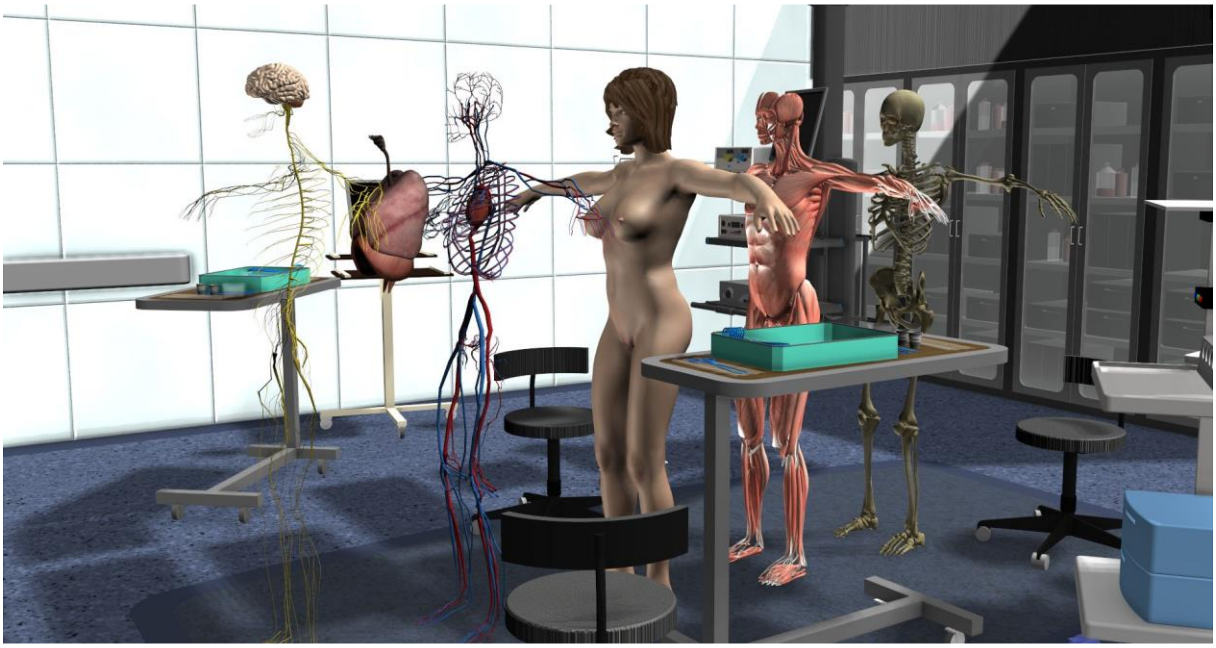

Figure 9. The virtual human body atlas in the exploded view

Initially, the application was made as an immediate solution, without use of any methodology. It was a basis for creation of KOMVR. After its creation, the application was experimentally re-created to verify effectiveness of KOMVR. The application was prepared twice by different teams, each time consisting of the same number of members with similar qualifications and skill level, as listed below:

- 2 virtual reality programmers
- 1 graphics designer (2D and 3D)
- 1 specialist – medical doctor

Selection of the hardware and software components was performed using different methods. The hardware (an HMD, tracking and gesture recognition device) was selected as a result of tests of many available solutions by a group of students (potential end users) and an evaluation study. Out of the possible devices, which included Samsung Gear VR and Oculus Rift HMDs, Microsoft Kinect tracking system, Myo armband, 5DT DataGlove, and a custom-made, dedicated low-cost mechanical tracking device with joystick, the Samsung Gear VR and Kinect were selected as projection and interaction devices, respectively, and were completed with a standard set of a monitor, mouse and wireless gamepad. The software was selected using the Analytical Hierarchy Process, which led to the discovery that the best possible solution was the EON Studio software, mainly because its ease of programming and availability (the university owned a commercial license, which was not the case for other considered software).

In the initial, non-methodological approach to this application, preparation of the application lasted 50 working days. Using the methodological approach – KOMVR – the time was shortened to 14 working days, with VR programming lasting approximately 8 hours. Proper acquisition and formalization of knowledge makes the programming process simple, as the formalized knowledge is ready for implementation in a VR engine. The application was positively evaluated by its end users both at a technical university and a medical university.

Other case studies

Currently, the authors are developing a number of other VR solutions for medical education. Among them, the following should be noted:

- presentation of an aortal valve replacement procedure in VR (**Figure 10**), application belonging to Level 2 (teaching procedural knowledge)
- “VR Surgeon”, an application for teaching behavior during lower limb amputation with use of VR and force feedback (**Figure 11**), belonging to Level 3 (teaching skills)

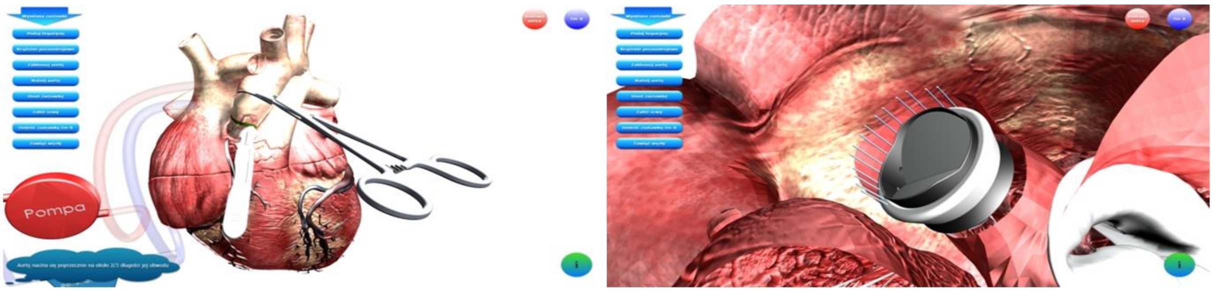

Figure 10. Aortal valve replacement in VR – example of application belonging to Level 2

Figure 11. Virtual lower limb amputation using haptic device – example of application belonging to Level 3

Both mentioned applications are complete in terms of development, but are still at the verification and correction stage before implementation. These applications are being prepared in cooperation with medical doctors from a medical university, for use by students during classes. KOMVR was applied to each of these cases, which allowed major time savings at the planning stages. The full effectiveness of KOMVR will be fully evaluated after the solutions are launched, but it can be said that gathering and formalizing the knowledge before the actual programming allowed this process to be shortened by 30-50%, with a decreased number of errors in comparison with medical education VR applications prepared earlier by the authors without any methodological approach (these are rough estimations based on labor consumption of the whole working team, spent on an application of comparable complexity).

CONCLUSION

The presented methodology of building KOMVR solutions is a result of years of experience of the authors' in the field of building interactive VR applications. Knowledge-engineering techniques were used to accelerate the process of planning, acquisition and formalization of knowledge, as well as to make the obtained VR solutions more effective in medical education. The effectiveness of KOMVR was confirmed by case studies and allows for considerable time savings in comparison with immediate solutions made without knowledge-engineering techniques. KOMVR was successfully used and fully implemented in practice in applications belonging to the basic level (presenting basic, general knowledge) and more advanced cases will be evaluated in further studies, as the authors plan to further develop this methodology. Both literature and practical studies have shown that VR is already an effective tool for medical education, and thanks to its low-cost solutions, will gain more and more popularity worldwide, improving or even replacing traditional teaching methods. The proposed methodology can make it easier for future VR creators to build educational applications, not only in medicine but also in other areas, such as engineering or general education.

REFERENCES

Abulrub, A. H., Attridge, A., & Williams, M. A. (2011). Virtual Reality in Engineering Education: The Future of Creative Learning. *International Journal of Emerging Technologies in Learning, 6(4)*, 4-11. doi:10.3991/ijet.v6i4.1766

Bell, J. T., & Fogler, H. S. (1995). The Investigation and Application of Virtual Reality as an Educational Tool. *Proceedings of the American Society for Engineering Education 1995 Annual Conference*. Retrieved from https://vrupl.evl.uic.edu/vrichel/Papers/ aseepap2.pdf

Biocca, F., & Delaney, B. (1995). Immersive virtual reality technology. In F. Biocca & M. R. Levy (Eds.), *Communication in the age of virtual reality* (pp. 57-124). Hillsdale, NJ: Lawrence Erlbaum Associates.

Bowman, D. A., & McMahan, R. P. (2007). Virtual Reality: How Much Immersion Is Enough? *Computer*, 40(7), 36-43. doi:10.1109/MC.2007.257

Buń, P., Górski, F., Wichniarek, R., Kuczko, W., Hamrol, A., & Zawadzki, P. (2015). Application of professional and low-cost Head Mounted Devices in immersive educational application. *Procedia Computer Science*, *75*, 137-146. doi:10.1016/j.procs.2015.12.235

Burdea, G. C., & Coiffet, P. (2003). *Virtual Reality Technology*. New York, NY: John Wiley & Sons, Inc.

Cruz-Neira, C., Sandin, D. J., & DeFanti, T. A. (1993). Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE. *SIGGRAPH '93 Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, 135-142. doi:10.1145/166117.166134

De Mauro, A., Carrasco, E., Oyarzun, D., Ardanza, A., Frizera Neto, A., Torricelli, D.,...Florez, J. (2011). Studies in Health Technology and Informatics. *Virtual reality system in conjunction with neurorobotics and neuroprosthetics for rehabilitation of motor disorders* (pp. 163-165). Newport Beach, CA: IOS Press.

Dingman, H. (2015). *The state of VR: Where Oculus Rift, HTC Vive, Gear VR, and others stand right now*. Retrieved from: CIO.com

Escobar-Castillejos, D., Noguez, J., Neri, L., Magana, A., & Benes, B. (2016). A Review of Simulators with Haptic Devices for Medical Training. *Journal of Medical Systems, 40(4)*, 104. doi:10.1007/s10916-016-0459-8

Getchell, K., Miller, A., Nicoll, J. R., Sweetman, R. J., & Allison, C. (2010). Games Methodologies and Immersive Environments for Virtual Fieldwork. *IEEE Transactions on Learning Technologies, 3(4),* 281-293. doi:10.1109/TLT.2010.25

Glatter, R. (2015). *How Virtual Reality May Change Medical Education and Save Lives*. Retrieved from: http://www.forbes.com/sites/robertglatter/2015/05/22/how-virtual-reality-may-change-medical-education-and-save-lives

Górski, F., Buń, P., Wichniarek, R., Zawadzki, P., & Hamrol, A. (2015). Immersive City Bus Configuration System for Marketing and Sales Education. *Procedia Computer Science, 75*, 137-146. doi:10.1016/j.procs.2015.12.230

Grajewski, D., Górski, F., Hamrol, A., & Zawadzki, P. (2015). Immersive and Haptic Educational Simulations of Assembly Workplace Conditions. *Procedia Computer Science, 75*, 359-368. doi:10.1016/j.procs.2015.12.258

Greuter, S., & Roberts, D.J. (2015). SpaceWalk: Movement and Interaction in Virtual Space with Commodity Hardware. *IE2014*, Newcastle, NSW. doi:10.1145/2677758.2677781

Gutierrez, M., Vexo, F., & Thalmann, D. (2008). *Stepping into Virtual Reality*. London: Springer-Verlag. doi:10.1007/978-1-84800-117-6

Hamrol, A., Górski, F., Grajewski, D., & Zawadzki P. (2013). Virtual 3D atlas of a human body - development of an educational medical software application, *Procedia Computer Science, 25*, 302-314. doi:10.1016/j.procs.2013.11.036

Jiang, M. (2011): Virtual Reality Boosting Automotive Development. In: Ma D., Gausemeier J., Fan X., Grafe M. (2011). *Virtual Reality & Augmented Reality in Industry* (pp. 171-180). Berlin, Springer Berlin Heidelberg. doi:10.1007/978-3-642-17376-9

Oculus Rift: 2014. Retrieved from http://www.oculusvr.com/

Oldham, K., Kneebone, S., Callot, M., Murton, A., & Brimble, R. (1998). MOKA-A Methodology and tools Oriented to Knowledge-based engineering Applications. In: N. Mårtensson, R. Mackay and S. Björgvinsson (Eds.), *Changing the Ways We Work* (pp. 198-207). Amsterdam, IOS Press.

Pandilov, Z., Milecki A., Nowak A., Górski F., Grajewski D., Ciglar D., Mulc T., & Klaić M. (2015). Virtual Modelling and Simulation of a CNC Machine Feed Drive System. *Transactions of FAMENA, 39(4)*, 37-54. Retrieved from: http://hrcak.srce.hr/file/223893

Reddy, E. J., Sridhar, C. N. V., & Rangadu, V. P. (2015). Knowledge Based Engineering: Notion, Approaches and Future Trends. *American Journal of Intelligent Systems, 5(1)*. 1-17. doi:10.5923/j.ajis.20150501.01

Riva, G. (2003). Applications of Virtual Environments in Medicine, *Methods of Information in Medicine 42(5)*. 524-534. Retrieved from http://cybertherapy.info/ MIM_0161_Riva.pdf

Scherman, W. R., & Craig, A. B. (2003). *Understanding Virtual Reality: Interface, Application, and Design*. Burlington, MA: Morgan Kaufmann.

Schreiber, G. (2007). Knowledge engineering. In V. Lifschitz, F. van Harmelen, and B. Porter, (Eds.), *Handbook of Knowledge Representation* (pp. 929-946). Amsterdam, Elsevier Science.

Seth, A., Vance, J. M., & Oliver, J. H. (2011). Virtual reality for assembly methods prototyping: a review. *Virtual Reality, 1(15)*, 5-20, doi:10.1007/s10055-009-0153-y

Stansfield, S., Shawver, D., Sobel, A., Prasad, M., & Tapia, L. (2000). Design and Implementation of a Virtual Reality System and Its Application to Training Medical First Responders. *Presence, IEEE, 9(6)*, 524-556. doi:10.1162/105474600300040376

Stokes, M. (2001). *Managing Engineering Knowledge: MOKA Methodology for Knowledge Based Engineering Applications*. London, Professional Engineering Publishing.

Tiwari, V., Jain, P. K., & Tandon, P. (2013). Design process automation KBE. *Proceedings of the World Congress on Engineering 2013 Vol. II.* Retrieved from http://www.iaeng.org/publication/WCE2013/WCE2013_pp737-742.pdf

Van Dam, A., Forsberg, A., Laidlaw, D. H., La Viola, J. J., & Simpson, R. M. (2000). Immersive VR for scientific visualization: a progress report. *Computer Graphics and Applications, IEEE, 20(6)*, 26-52. doi:10.1109/38.888006

Vankipuram, A., Khanal, P., Ashby, A., Vankipuram, M., Gupta, A., Drumm Gurnee, D., Smith, M. (2014). Design and Development of a Virtual Reality Simulator for Advanced Cardiac Life Support Training. *Journal of Biomedical and Health Informatics, IEEE, 18(4),* 1478-1484. doi:10.1109/JBHI.2013.2285102

Verhagen, W. J. C., Bermell-Garcia, P., Van Dijk, R. E. C., & Curran, R. (2012). A critical review of Knowledge-Based Engineering: An identification of research challenges. *Advanced Engineering Informatics, 26(1)*. 5-15. doi:10.1016/j.aei.2011.06.004

Zawadzki, P., Górski, F., Kowalski, M., Paszkiewicz, R., & Hamrol, A. (2011). A system for 3D models and technology process design. *Transactions of FAMENA, 35(2),* 69-78. Retrieved from http://eds.a.ebscohost.com/eds/pdfviewer/pdfviewer?vid=3&sid= dd364f39-f901-4ae7-9e14-d97f0fbe07f5%40sessionmgr4004&hid=4113

Zhaoliang, D. (2011). 3D tracking and position of surgical instruments in virtual surgery simulation. *Journal of Multimedia, 6(6)*, 502-509. doi:10.3233/978-1-60750-706-2-581

http://iserjournals.com/journals/eurasia