Development of Augmented Reality Training Simulator Systems for Neurosurgery Using Model-Driven Software Engineering

Conference Paper · April 2017

DOI: 10.1109/CCECE.2017.7946843

4 authors, including:

Hamza Ghandorh
Taibah University

Sandrine De Ribaupierre
The University of Western Ontario

Roy Eagleson
The University of Western Ontario

2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)

Development of Augmented Reality Training Simulator Systems for Neurosurgery Using Model-Driven Software Engineering

Hamza Ghandorh, Justin Mackenzie, and Roy Eagleson
Electrical and Computer Engineering Department,
Western University,
London, N6A3K7, Canada
Email: $\{hghandor\}\{jmacke26\}\{eagleson\}$@uwo.ca

Sandrine de Ribaupierre
Department of Clinical Neurological Sciences,
Western University,
London, N6A3K7, Canada
Email: sderibau@uwo.ca

***Abstract*—Neurosurgical procedures are complicated processes, providing challenges and demands ranging from medical knowledge and judgment to the neurosurgeons dexterity and perceptual capacities. Deliberate training of common neurosurgical procedures and underlying tasks is extremely important. One effective method for the training is to enhance the required surgical training tasks through the use of neurosurgical simulators. Development of neurosurgical simulators is challenging due to many reasons. In this work, we proposed to facilitate the development of new augmented reality neurosurgical simulator systems through the adoption of model-driven engineering. Our developed systems involve the interactive visualization of three-dimension brain meshes in order to train users and simulate a targeting task towards a variety of predetermined virtual targets. We present our results in a way which highlights two new design artifacts through our MDE approach.**

***Index Terms*—model-driven engineering; augmented reality; neurosurgery; simulator; targeting task; external ventricular drain; Unity3D; Vuforia;**

I. Introduction

Neurosurgical procedures are complicated processes that include a combination of neurosurgical operations and overlapping surgical tasks. When a neurosurgeon performs a surgical task (e.g. manipulation of medical instruments towards a specific target), it is important to minimize damage to healthy tissues and eloquent brain structures. A potential risk of damaging eloquent brain areas is unavoidable when trainees make multiple hits towards a brain structure by which a set of undesirable trajectories are created. Because the neurosurgery occupation necessitates an education that is as practical as the occupation itself [1], deliberate training of common neurosurgical procedures and underlying surgical tasks, therefore, is important. An effective method for deliberate training is to replicate the required procedures or tasks through the application of neurosurgical simulators, visualizing three-dimension (3D) medical imaging data (e.g., magnetic resonance imaging (MRI) and X-ray computed tomography (CT)).

Development of a neurosurgical simulator is non-trivial due to many reasons. First, effective neurosurgical simulators have complex requirements specifications. For example, the neurosurgical simulator uses extensive equipment, and need setup and technical expertise. Many technologies (image registration, rendering of the medical imaging data, object tracking) are utilized to fulfill the design requirements [2] [3]. Non-functional requirements (system scalability, extensibility, and modifiability) and the human factors aspects (human perceptual, motor, and cognitive capacities) are important factors that impact performance in the use of the simulator. Usability testing in this regard provides us with estimates of the validity of neurosurgical simulators [4] [5]. Designing of neurosurgical simulators requires a comprehensive approach to development. Otherwise, design and development resources are wasted in trial-and-error simulator prototypes for accurate representing of real case neurosurgical settings. Our approach is to begin by modeling the surgical scenarios using rich schemes structured around Hierarchical Task Analysis (HTA) [6]. HTAs are represented as formal abstract models that describe the simulators workflow and will direct the design, implementation, and evolution of future neurosurgical simulators at the abstract level. The more the surgical tasks structure changes, the more the design and the implementation of simulators artifacts need to adapt to enforce longevity of the simulators. These schemes are then represented in the software development process using Harel Statecharts, which can model abstract hierarchies of states which can have orthogonal substates.

Accordingly, we propose this as software development model and in the sequel, we demonstrate two case studies in which augmented reality (AR) training simulator systems for neurosurgical targeting tasks through the adoption of Model-driven Engineering (MDE). The purpose of systems is to visualize 3D human brain meshes in order to train users and simulate a targeting task towards a variety of predetermined virtual targets. We have a hypothesis that MDE process will facilitate a quick production of AR training simulator systems within reasonable development costs.

The outline of this paper is organized as follows: Section II introduced relevant background for our project, Section III describes our approach towards the development of the systems, Section IV indicates our current results in terms of generated artifacts, Section V of this paper indicates work related to our

978-1-5090-5538-8/17/$31.00 ©2017 IEEE

systems, and Section VI concludes our paper.

II. Background

A. MDE process

MDE is a software development process that entails the application of abstract models and modeling technologies to simplify and formalize various software development activities [7]. MDE promotes model transformations as the primary activity prior to producing executable artifacts with domain constraints or rules. MDE process aims to hide complex platform details, to reduce manual coding errors, to boost developers' productivity, to improve quality, and to share experts' knowledge in a specific domain [8].

Several approaches have been proposed to represent the view of MDE, such as model-driven architecture (MDA), agile MDE, domain-specific programming, or software factories [7]. MDA is a common approach for MDE. MDA separates software development into two application-oriented models: platform-independent models (PIM) and platform-dependent models (PSM). PIM are reusable cross-platform models that do not contain a reference to any platform, while PSM models are a tightly coupled version of PIM, built towards a certain platform. The syntax of PSM will be the input for code generators, aims to source code production [8].

B. AR in medical settings

AR technology provides a real-time view of the physical world that has been superimposed with virtual computer-generated information [9]. In neurosurgery context, AR visualizes spatial and anatomical information about a patient's case that were not available before the surgery. The use of AR systems for simulation purposes provides many advantages for neurosurgeons in terms of visualization and training aids where trainees train for either full procedures or a specific surgical task within a predetermined scenario. AR applications could support trainees to cultivate their surgical skills, to evaluate their performance, to reduce surgical errors, and to reduce the need for a larger incision.

C. External ventricular drain (EVD) placement task

EVD placement task is a necessary skill for a set of neurosurgical procedures, such as Ventriculostomy[1]. Figure 1 depicts an EVD placement task during the Ventriculostomy context. EVD placement task - a targeting task- aims to choose an appropriate burr hole on the skull and blindly place a catheter through the burr hole to intersect a lateral ventricle in order to drain cerebrospinal fluid and relieve intracranial pressure. EVD placement task is an essential surgical skill in emergency cases, and medical trainees are required to develop such skill as part of their training curriculum.

[1]Ventriculostomy is a neurosurgery that concerns intrusion into a brain's ventricle.

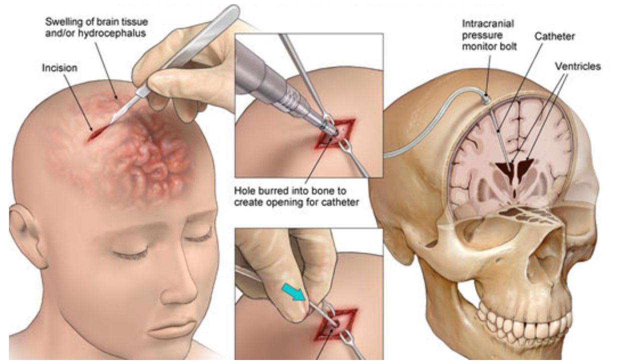

Fig. 1: An overview of Ventriculostomy neurosurgical procedure. An EVD placement task is also presented in the middle lower part [10].

III. Our model-driven engineering approach

We have developed two new AR systems to support novice medical trainees to cultivate the skill of EVD placement tasks towards a specific target with a given neurosurgical scenario. The first AR system will target desktop machine for open surgery simulations, and the second AR system will target mobile machine for immersive surgery simulations.

A. Requirement specifications

Our requirements specifications entail a typical neurosurgical scenario and a few design considerations. Senior neurosurgeons verbal description, neurosurgical scenarios repository and neurosurgical education materials are some contributing factors to maintaining neurosurgical scenarios. Several concepts in mind developed the description of the neurosurgical scenario: who are the actors in the scenario? what is the task being performed? how are the tasks decomposed into child tasks? what are the events that cause an actor to complete one task and move onto another ? what objects are associated in the scenario? what is the metrics to be measured during the scenario execution?

The neurosurgical scenario included a sequence of surgical tasks within a neurosurgical procedure, and the description of EVD placement tasks (see section II-C). Each EVD placement task decomposed to the lowest child level. The goal of the task decomposition is to break down the task to the point where the lowest level child tasks are simple tool movement tasks; this approach will allow for simpler and more reliable performance evaluation techniques rather than attempting to perform an evaluation on the more complex and higher level tasks. Another benefit to this approach is that the evaluation of individual tasks allows for more constructive data and feedback from users. For our purpose, our neurosurgical scenario represented as surgical HTA bullet-points format. A portion of our neurosurgical scenario is presented as follows:

- Phase 0: review imaging studies (CT scans or MRI).
- Phase 1: localize the closest entry point in the right frontal bone in the skull to the lateral ventricles.
- Phase 2: visualize the lateral ventricles shape and location in your mind.

- Phase 3: orient your EVD to be in line with the best trajectory.
- Phase 4: advance the EVD towards the lateral ventricles until you hit the frontal horn.
  1) Penetrate the EVD through the inner table of the skull bone perpendicular to the brain surface slowly to <= 6 cm depth.
  2) The stylet then is removed and the catheter is kept in the burr hole.
  3) A slight pop in resistance emerged (i.e. increase resistance than a loss of resistance).
  4) When the frontal horn in cannulated, CSF drainage should emerge through the catheter.

For simplicity and evaluation of users performance, we focused on Phase 4.1 from the scenario that fulfills the required targeting task. In addition, there are a few design considerations for our systems. The systems need to visualize 3D brain structures on AR display modalities (i.e. AR scene running on a screen). The systems allow a mode for users to perform EVD placement tasks towards salient targets in the virtual space. The systems need also to be deployable on a different platform and to be extendable for more features. Moreover, the systems need to allow a mode for evaluators to insert neurosurgical scenarios and setup evaluations sessions by recording task time and the targeting effect within the virtual space.

B. Methodology

Figure 2 indicates an overview of our MDE approach. We have adopted MDA process where we considered the model transformation between PIM and PSM. In order to fulfill our goal, we followed the following steps. Firstly, we identified a set of requirements specifications (see section III-A)(Figure 2.1). The neurosurgical scenario was represented in a Harel statechart indicating users, system states, and events workflow within the neurosurgical scenario, where it was represented in XML format. Secondly, we specified visual representations of the EVD placement tasks and verified them with a senior neurosurgeon during the designing of both systems. The visual representation is a few components, such as 3D brain mesh, ellipsoids, and cylinder, of an interactive AR scene within Unity3D[2] editor. We considered the components of interactive AR scene within Unity3D as PIM. PIM was created based on XML syntax, obtained from the neurosurgical scenario. Whenever the neurosurgical scenario is developed, the scenario and its related data were stored through a scenario management service, a RESTful HTTP web service in the cloud which stores data into a SQL server database. The RESTful web service was used to store, modify and query scenarios; client applications were developed for providing a user interface for managing these scenarios and the simulator itself queries and retrieves the scenario such that it can initialize and setup the simulator state and scene based on the scenario. The scenario management service facilitated the transformation between the XML tags to the Unity3D scene (Figure 2.2). Whenever the AR scene is completed, we used the Unity3D editor to assign the AR scene settings, required to the generated artifices and needed platforms (i.e. desktop, mobile) (Figure 2.3). We verified the generated artifacts against the requirement specifications. Then, the AR scene created PSM prior to being transformed to the code generation step. Thirdly, the code generator the Unity3D engine - got PSM syntax for both systems and transformed it to the source code, and the Unity editor deployed the source code towards the targeted platforms (i.e. Macbook Pro machine, Android LG Nexus 5 smartphone). We produced the two artifacts with several prototypes of each system to be validated by the senior neurosurgeon in order to check how realistic they are (Figure 2.4). Lastly, we asked a user to use both artifacts and to perform the EVD placement tasks according to the neurosurgical scenario. Once the user begins to perform the tasks, the simulator will be collecting the events performed by the user and also be measuring various metrics such as tool placement, viewpoint, time, etc (Figure 2.5). With this approach, given the starting state of the scenario, the evaluator of the performance can reconstruct the state of the performance at any point within the performance, this allows the evaluator to replay the performance and analyze the performance in a very robust manner. Once a number of EVD placement tasks have been completed, the EVD placement tasks events will be analyzed.

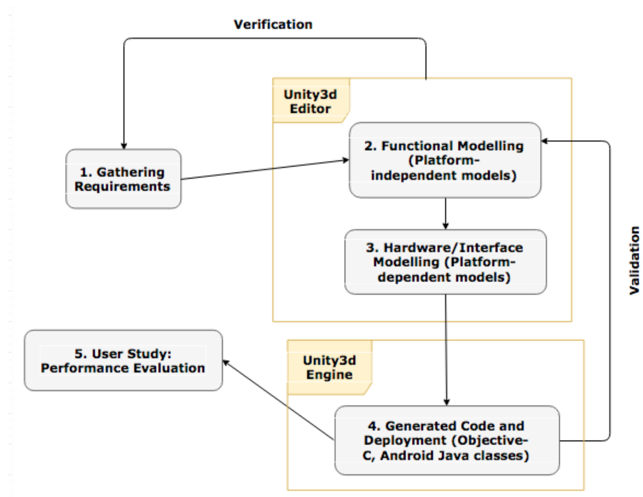

Fig. 2: A simplified flowchart of applied MDE.

C. Validation

User performance evaluation is a key aspect for the validity of any solution in the context of neurosurgical simulations. The evaluation of trainees' performance usually focuses on their surgical competency through psychomotor tasks (i.e. a manipulation of physical objects). Since any surgical task involves from the navigation of a tool to manipulation of

[2]Unity3D is a closed sourced game engine that concerns not only creating interactive game scenes with AR functionality but also allowing to texture and configure 3D meshes for the virtual world. Unity3D site https://docs.unity3D.com/Manual/index.html

tissue at a specific location, our approach for the evaluation of trainees performance should involve a 3D extension of Paul Fitts methodology [11]. Fitts investigated performance in human psychomotor behavior (i.e. moving an object from one box to another) through a variety of one-degree-of-freedom tasks, extending Shannons theorem 17 in information theory. Fitts Law allows to predict human movement and to determine human performance as a trade-off between accuracy and speed to determine the index of performance (IP) (bit\second), which is measured as Equation 1.

$$IP = \frac{ID}{MT} = \frac{\log_2(\frac{A}{W}+1)}{MT} \tag{1}$$

MT is movement time (seconds) needed to accomplish a task, index of difficulty (ID) is how accurate and fast the participants were to complete the trials (bits), W represents targets width (mm) and A is the distance between the starting point of the user and the center of the targets (mm). Fitts law states that mean time to perform a movement and selection task is logarithmically related to the width of the target and the distance to the target. The inverse of ID, the logarithmic relation of W and A, represents IP. IP represents how well a user performed the movement task, which leads to a shorter time to perform a given task of varying difficulty.

D. Materials

We used several closed-source commercial-off-the-shelf software packages: Unity3D engine (version 5.3), Vuforia[3] SDK (version 6), and 3D meshes needed to visualize targets and brain biomedical image dataset (i.e. 3D skull and brain structures) [12]. The AR functionality was written in C# programming language. The 3D brain meshes are depicted in Figure 3; these meshes were designed by Allen et al. [12] and used for educational purposes. The 3D brain meshes designed through Blender[4], 3D modeling and animation suite, and they present a comprehensive representation of human brain structures needed for the targeting tasks.

Both artifacts needed two fiducial markers, which depicted at Figure 4. The two traceable fiducial markers used to reconstruct virtual objects in the AR scene: (4.a) triangles marker is responsible for rendering the 3D brain meshes (human skull and the brain structures), and (4.b) pebbles marker is responsible for rendering a 3D surgical tool mesh (EVD catheter).

IV. Results and discussion

We were able to produce the two systems through the adoption of MDA process. The purpose of the systems is to visualize 3D human brain meshes in order to train users and simulate the targeting tasks towards a variety of predetermined virtual targets.

[3]Vuforia is a tracking and detection engine that provides object detection and tracking functionalities and reconstructing virtual environments, plus it is responsible for the registration and deployment process of the fiducial markers, image processing, and camera calibration. Vuforia site: https://library.vuforia.com/all-articles

[4]Blender site: https://www.blender.org/manual/

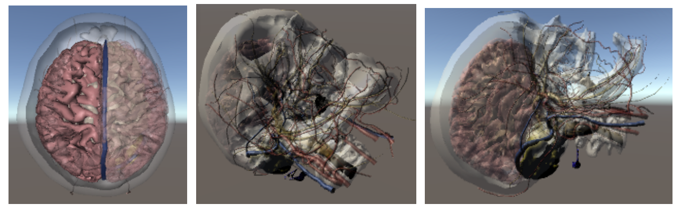

Fig. 3: Different views of the virtual 3D brain meshes on which targeting tasks were performed in the virtual world [12].

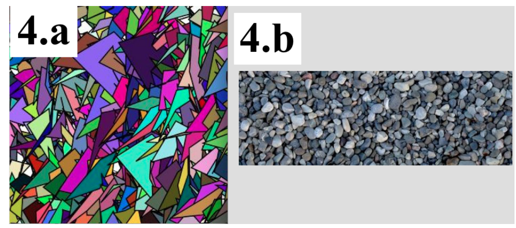

Fig. 4: The two traceable fiducial markers.

A. Generated artifacts

An overview of the both systems is depicted in Figure 5 and Figure 6, respectively. The first and the second generated artifact deployed in the following environments, described in Table I. In Figure 7, Figure 8, and Figure 9, our artifacts were deployed and running through a validation stage.

TABLE I: The generated artifact environments

Generated artifacts	Enviroment	Technical specifications
First system	(5.a) 32″ screen	RCA TV, 1366x768 c, 16:9 widescreen.
	(5.b) external camera	Logitech HD Webcam C270.
	(5.c) and (5.d) two traceable fiducial markers	Used images have sharp and spiked features and chiseled texture, are rich in details, have good contrast, and with 8 or 24 bit, RGB or greyscale format.
	MacBook Pro laptop machine	OS X El Capitan10.11 operating system, 2.66 GHz Intel Core 2 Duo processor and 8GB 1067MHz DDR3 Ram.
	Cardboard cylinder	Diameter x height (5.6 x 27)(cm).
Second system	(6.a) Google Cardboard DSCVR Headset	Two 34 mm biconvex lenses, fits for 5.96″,screen phones, black color.
	(6.b) LG Nexus 5 smartphone	4.95″ screen (1080x1920 resolution, (137.9 x 69.2 x 8.6 mm), Android v6.0 (Marshmallow) operating system, Quad-core 2.3 GHz Krait 400 processor, camera 8 MP).

In order to have a realistic position of the 3D brain meshes superimposed on fiducial markers, Unity3D gave us the ability to adjust the 3D meshes location (X axis, Y axis, Z axis), rotation (rolls, pitch, yaw), and scale values

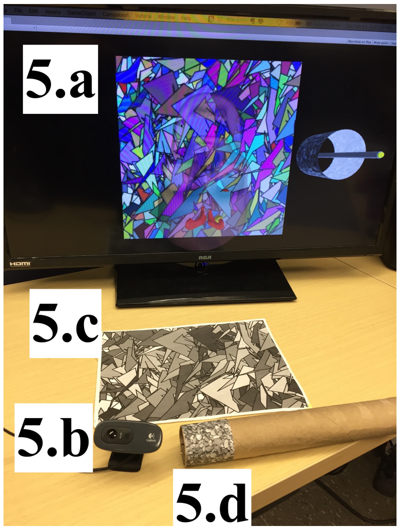

Fig. 5: The environment of the first generated artifact.

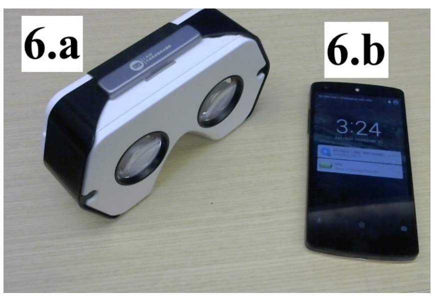

Fig. 6: The environment of the second generated artifact.

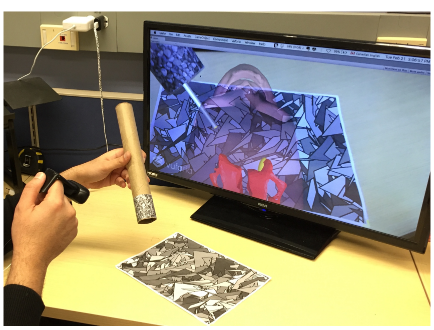

Fig. 7: A user is performing an EVD placement task using the first generated artifact, deployed in the Macbook pro machine and shown through the 32″ screen. The 32″ screen an AR scene that includes an EVD target (yellow ellipsoid), which are predefined entry points to the lateral ventricle mesh (i.e. red virtual 3D object) and the patient skull (transparent purple virtual 3D object).

Fig. 8: The user is performing the EVD placement task using the second generated artifact, deployed in the LG Nexus 5 smartphone and shown through the Google cardboard headset.

Fig. 9: The AR scene is running through Google Cardboard headset when the user performed the EVD placement tasks. The headset is presenting an immersive view of the AR scene.

before\post running the prototypes. Whenever we changed the configurations of components of the AR scene, Unity3D gave a visual feedback of the current state of these components, accordingly. Although the integration of the Unity3D engine and the Vuforia sdk is a wise chain of tools to implement into AR applications, we could not have an accurate augmentation of the 3D brain meshes on the fiducial markers. We noticed this phenomenon within the user wanted to move around the fiducial markers for 180° views prior to figure out a best possible trajectory towards their targets. This issue affects the ability of the user to determine the depth of the designed targets when the 3D brain meshes. Although many AR systems have been developed in recent years, we believe the simplicity of our design is a valuable contribution that makes such designs more available to a wider range of users.

B. Initial usability results

A targeting task experiment was conducted to investigate the usability of the first generated artifacts. The experiment intended to determine participants' performance with a variety of ellipsoids with different location, rotation, and size (i.e. ellipsoids widths are 0.10, 0.15, 0.05 mm) and from different distances. Seven novice graduate students, with no background in anatomy or simulation, were asked to join our experiment by introducing a virtual surgical tool and navigating towards an existing target on a virtual human brain model in the virtual world. Each participant performed 27 trials for 27 ellipsoids (i.e. 189 trial in total) through three sessions where participants completed their trials in random order. Average of MT of participants is presented in Figure 10. The current results seem promising, yet a further investigation will be conducted.

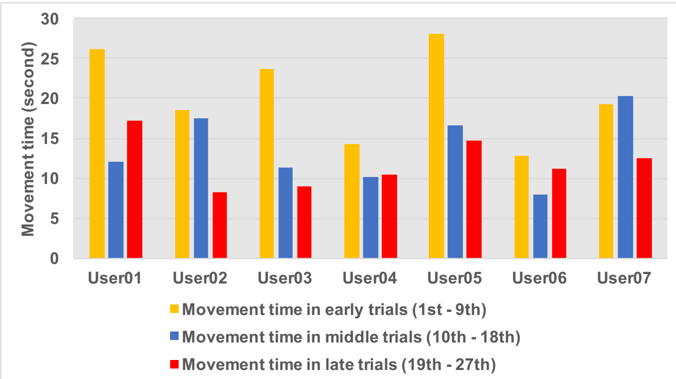

Fig. 10: Average of completion time (i.e. movement time (MT)) changes between early, middle, and late trials.

V. Related work

There are few attempts to develop neurosurgical simulator systems based on abstract models, yet they did not explicitly follow any MDE approaches. Jannin et al. [13] proposed an imaged-guided system for craniotomy procedures based on live surgical observation, and they used a UML class diagram models to represent their surgical scenarios. Choudhury et al. [14] proposed a conceptual framework intended as the base to develop neurosurgical oncology modules for training purposes. Claude [15] proposed a methodology to develop surgical procedural simulators based on generating generic abstract models.

Other related work utilized MDE process to generate their medical artifacts through a case study format. Khriss and Mckibben [16] reported two case studies to generate two medical systems, ePetrass, and SRADC. Onisto et al. [17] produced their cross-platform B-mode ultrasound imaging system for two different platforms. Both works did not aim for the development of AR training simulators for neurosurgery context.

VI. Conclusion and future work

Detailed case studies in MDE are rare, especially for the development of neurosurgery training simulators. In this work, we presented our efforts to develop two AR task training simulator systems based on MDE process. Currently, we are using our prototypes for further user studies to measure the performance of novice users during their targeting tasks. The performance management service is completed and allows for performance data to be stored and retrieved for analysis later on. The used neurosurgical scenarios are simple, and they do not include plausible neurosurgical complications, as for instance, bleeding when performing the EVD targeting tasks. In addition, we will investigate the user performance with application of depth clues in the virtual world when performing the EVD targeting tasks.

Acknowledgements

Thanks to CSTAR[5]. The research for this work was financially supported by Taibah University, Madinah, Saudi Arabia[6], and SACB[7].

References

[1] B. D. Johns, "The creation and validation of an augmented reality orthopaedic drilling simulator for surgical training," Ph.D. dissertation, 2014.
[2] M. Kramers, R. Armstrong, S. M. Bakhshmand, A. Fenster, S. de Ribaupierre, and R. Eagleson, "Evaluation of a mobile augmented reality application for image guidance of neurosurgical interventions." *Studies in health technology and informatics*, vol. 196, pp. 204–208, 2014.
[3] Q. Zhang, R. Eagleson, and T. M. Peters, "Real-time visualization of 4d cardiac mr images using graphics processing units," in *3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2006.*, April 2006, pp. 343–346.
[4] R. Armstrong, "An examination of factors that impact performance on a patient-specific VR-based ventriculostomy simulator," in *Proceedings of the 29th International Congress and Exhibition Computer Assisted Radiology and Surgery*. Barcelona, Spain: CARS, June 24–27 2015.
[5] K. Abhari, J. S. H. Baxter, E. C. S. Chen, A. R. Khan, T. M. Peters, S. de Ribaupierre, and R. Eagleson, "Training for planning tumour resection: augmented reality and human factors," *Biomedical Engineering, IEEE Transactions on*, vol. 62, no. 6, pp. 1466–1477, Jun. 2015.
[6] S. Sarker, A. Chang, T. Albrani, and C. Vincent, "Constructing hierarchical task analysis in surgery," pp. 107–111, 2008.
[7] B. Hailpern and P. Tarr, "Model-driven development: The good, the bad, and the ugly," *IBM Systems Journal*, vol. 45, pp. 451–461, 2006.
[8] S. J. Mellor, A. N. Clark, and T. Futagami, "Model-driven development - Guest editor's introduction," *IEEE Software*, vol. 20, pp. 14–18, Sep. 2003.
[9] R. T. Azuma, "A survey of augmented reality," *Presence*, vol. 6, pp. 355–385, 1997.
[10] N. Holland, *Neurology Update blog*, [Online]. Available: https://mmcneuro.files.wordpress.com/2013/03/ventriculostomy.png?w=600. [Accessed: 26- Oct- 2016]., 2016.
[11] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement." *Journal of Experimental Psychology*, vol. 47, no. 6, pp. 381–391, 1954.
[12] L. K. Allen, H. Z. Ren, R. Eagleson, and S. de Ribaupierre, "Development of a Web-Based 3D Module for Enhanced Neuroanatomy Education," vol. 220, no. 22, 2015, pp. 5–8.
[13] P. Jannin, M. Raimbault, X. Morandi, and B. Gibaud, *Modeling Surgical Procedures for Multimodal Image-Guided Neurosurgery*. Springer Berlin Heidelberg, 2001, pp. 565–572.
[14] N. Choudhury, N. Gélinas-Phaneuf, S. Delorme, and R. Del Maestro, "Fundamentals of Neurosurgery: Virtual Reality Tasks for Training and Evaluation of Technical Skills," *World Neurosurgery*, vol. 80, pp. e9–e19, Nov. 2013.
[15] G. Claude, V. Gouranton, B. Caillaud, B. Gibaud, B. Arnaldi, and P. Jannin, "Synthesis and Simulation of Surgical Process Models." *Studies in health technology and informatics*, vol. 220, pp. 63–70, 2016.
[16] I. Khriss and A. McKibben, "Accelerating the Development of Health and Social Services Systems through Model-Driven Engineering: Two Case Studies," in *2015 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, Dec. 2015, pp. 726–731.
[17] H. J. Onisto, T. de Moraes Machado, R. C. Fernandes, J. D. de Medeiros, I. Tamagno, T. C. Dezotti, and J. E. Bertuzzo, "Model-driven engineering applied to the development of embedded software for B-mode ultrasound imaging systems - A case study," in *2014 IEEE International Ultrasonics Symposium*. IEEE, Sep., pp. 1261–1264.

[5]CSTAR site: http://www.lhsc.on.ca/About_Us/CSTAR/
[6]Taibah University site: https://www.taibahu.edu.sa/Pages/EN/Home.aspx
[7]SACB site: http://www.saudibureau.org/en/