

LISSA: A Serious Game to learn Cardiopulmonary Resuscitation

Wattanasoontorn V.*
Graphic and Image Laboratory
Institute of Informatics and
Applications,
University of Girona, Spain
voravika@gmail.com

Boada I.
Graphic and Image Laboratory
Institute of Informatics and
Applications,
University of Girona, Spain
imma.boada@udg.edu

Sbert M.
Graphic and Image Laboratory
Institute of Informatics and
Applications,
University of Girona, Spain
mateu@ima.udg.edu

ABSTRACT

Cardiopulmonary resuscitation (CPR) is a first aid key survival technique to recover victims from sudden cardiac arrest. In this paper we present the framework of LISSA which is a serious game designed to teach and learn CPR skills. LISSA combines in a single framework computer-based simulations of CPR emergencies with the functionalities of e-learning platforms. Using UML state machine, different emergency scenario can be derived by instructor and presented as problems that the learner has to solve in a game mode. All the actions and all the cases are registered in a database that can be consulted at any moment by the instructor. LISSA can be used as a substitute or a complement for traditional CPR classroom-based instruction. It can also be used to educate laypersons who might not otherwise be inclined to attend a CPR course. The adaptive learning supported by LISSA makes it suitable to refresh and improve CPR skill retention over time.

Keywords

Life Support, Serious Games, Medical Simulation, Cardiopulmonary Respiration, Resuscitation

1. INTRODUCTION

Cardiopulmonary resuscitation (CPR) is a first aid key survival technique used to stimulate breathing and keep blood flowing to the heart. Effective CPR administration can significantly increase the chances of survival for victims of cardiac arrest that take place outside of hospital. Since 1960, when Kouwenhoven published an article stating that anyone, anywhere, could perform CPR [13], providing CPR has become an essential competency not only for expert or professional but also for laypersons.

*Correspondant Author

CPR protocol consists in a set of procedures that have to be applied in a correct order and in an specific way. These procedures determine, among others, how to assess an unconscious person by checking the Airway, Breathing and Circulation (known as ABCs, [10]) and how to do chest compression. To learn CPR different strategies have been proposed, traditional classroom, video self-instruction and computer-based programs [2, 3, 5, 9, 19, 22].

In this paper we present LISSA (Life Support Simulation Application), a serious game designed to teach and learn CPR. LISSA exploits video game technology to link in a single framework computer-based case simulations with e-learning functionalities.

The paper has been structured as follows. In Section 2 previous work on CPR and serious games is presented. In section 3 we describe LISSA architecture and the main modules that compose it. In Section 4 results and implementation details are given. Finally, in Section 5, conclusions and future work are presented.

2. RELATED WORK

In this section, we describe the basic CPR protocol and also previous work on applications and serious games designed to learn CPR.

2.1 The basic CPR flowchart

The CPR procedures that have to be applied in an emergency case depend on different parameters such as the age of the patient, his initial situation, or the presence of medical devices. To know how to proceed in each case we can follow some of the graphic flowcharts provided by different organizations such as the European Resuscitation Council (ERC) [26], the Red Cross or the American Heart Association [2]. In these sites, we can found from the basic CPR flowchart to the more advanced one.

As an example, consider an emergency case where an adult patient is lying on the floor. To perform CPR we proceed as follows. To start, the helper checks if the patient is conscious by checking gently or shouting loudly. If the patient responds the helper can consider that the patient is fine and then only has to observe for new symptoms. If no symptoms appear the situation is controlled and no help is necessary. If the patient does not respond, the helper has to open the airway in order to give an easy posture to get air circulation.

| No. | Name | Year | Platform | Interaction tool | Interface | Connectivity | Purpose |
|-----|-----------------------|------|----------|------------------|-----------|--------------|-------------------|
| 1 | JUST | 2003 | PC | Mouse | 3D | Online | Learning |
| 2 | CPR simulator | 2006 | Handheld | Mouse | 2D | Offline | Testing |
| 3 | M-AID | 2007 | Handheld | Mobile Button | 2D | - | Testing |
| 4 | iCPR | 2009 | Handheld | Accelerometer | 2D | Online | Testing |
| 5 | iResus | 2010 | Handheld | Touch Screen | 2D | Online | Testing |
| 6 | AED Challenge | 2011 | PC | Mouse | 2D | Online | Learning |
| 7 | CPR & Choking | 2011 | Handheld | Touch Screen | Video | - | Learning |
| 8 | Staying alive | 2011 | PC | Mouse | 3D | Online | Testing |
| 9 | CPR Game | 2012 | Handheld | Touch Screen | 2D | Online | Testing |
| 10 | MicroSim-Prehospital | 2012 | PC | Mouse | 2D, Video | Offline | Learning, Testing |
| 11 | Mini-VREM | 2012 | PC | Kinect | 2D | Offline | Testing |
| 12 | Save-A-Life Simulator | 2012 | PC | Mouse | Video | Online | Learning |
| 13 | LISSA | 2013 | Multi | Mouse, Kinect | 3D | Online | Learning, Testing |

Table 1: Comparison of CPR applications
(-) represents that this feature is not mentioned in the reference

In this state, breath is observed. If the patient is breathing normally, the helper should arrange the patient posture to the recovery one to avoid suffocation and observe until patient is fine. If the patient is not breathing, the emergency call is urgently required and a sequence of chest compression and breath delivery maneuver has to be applied until help arrives.

2.2 Serious games and CPR

Different applications and serious games to learn CPR have been proposed in last years. Among them, there are video training applications such as *Save-A-Life Simulator* [18], an interactive online video simulation that tests the player's knowledge of helping someone suffering from sudden cardiac arrest; and *CPR & Choking* [25], an application that provides instant information on how to perform CPR and how to aid a choking victim. These applications show a one minute video to present the latest recommendations from the major international resuscitation organizations including the American Heart Association and the International Liaison Committee on Resuscitation. There are also handheld applications such as, *CPR Game* [8], a cardiac arrest simulator on iOS platform focused on advanced CPR training guided by 2010 Advanced Cardiovascular Life Support guidelines; *iResus* [16], an application for smart phone, designed to improve the performance of an advanced life support provider in a simulated emergency situation; *iCPR* [6] [24], an iPhone application designed for both lay persons and healthcare professionals able to detect the rate of chest compressions performance by using the built-in accelerometer; *M-AID* [28], a first aid application for mobile phones that uses *yes* or *no* questions to judge an ongoing situation giving to the user detailed instructions of how to proceed; and *CPR simulator* [15] [17], a set of CPR exercises including adult, child and infant CPR simulator that runs through the CPR sequence.

In addition, some applications for PC platforms are *Mini-Virtual Reality Enhanced Mannequin (Mini-VREM)* [23] [12] which is a CPR feedback device with motion detection technology including Kinect, sensor and software specifically designed to analyse chest compression performance and provide real-time feedback in a simulation training setting, and *AED Challenge* [1], an application that provides online automated external defibrillation and CPR skill practice and testing with realistic scenarios. Finally, in the serious games

context, some games for CPR training are *JUST* [21], an immersive VR situation training system for non-professional health emergency operators, *MicroSim Prehospital* [14] designed for pre-hospital training on emergency medical services, and *Staying alive* (2011) [7], an online 3D simulator which provides a learning experience of saving a virtual patient from cardiac arrest in four minutes.

Table 1 presents main features of all reported methods. From left to right we present the year of publication, the platform for which has been designed, the interaction tool that supports, the type of interface, distinguishing between 2D, 3D and video, the required connectivity and also the main purpose of the application. In the last row we also include the features of our proposed framework LISSA, which is 3D multi-platform and supports interaction via mouse and kinect. The platform works online and its main objective is learning and testing. One of the LISSA has been conceived considering both teachers and learners and with the idea to make teachers tasks easier. In this way, LISSA allows the creation of different scenarios with different characters, patient symptoms and environments. Moreover, the system provides automatic feedback to the learner which enhances the learning process.

3. LISSA OVERVIEW

LISSA has been designed to introduce and increase the knowledge of CPR to any kind of public. In this section, we present the main components of this game, how we define the CPR scenarios and also the score strategy.

3.1 The main components

LISSA has been designed as an e-learning environment with all the actions turning around a CPR scenario that reproduces with 3D realism an emergency situation that requires CPR procedures. LISSA supports two type of users, instructors and learners. The CPR scenario is defined by the instructor and presented to the learner as a test or problem. The learner solves the problem applying the CPR procedures in a game mode. LISSA evaluates the actions and assigns a final score. All the learner's actions are registered in a central data base allowing instructors to consult them in order to track the learning process. Instructors can use this information to recommend new scenarios and problems. The main components and functionalities of the LISSA framework are presented in Figure 1 and described below.

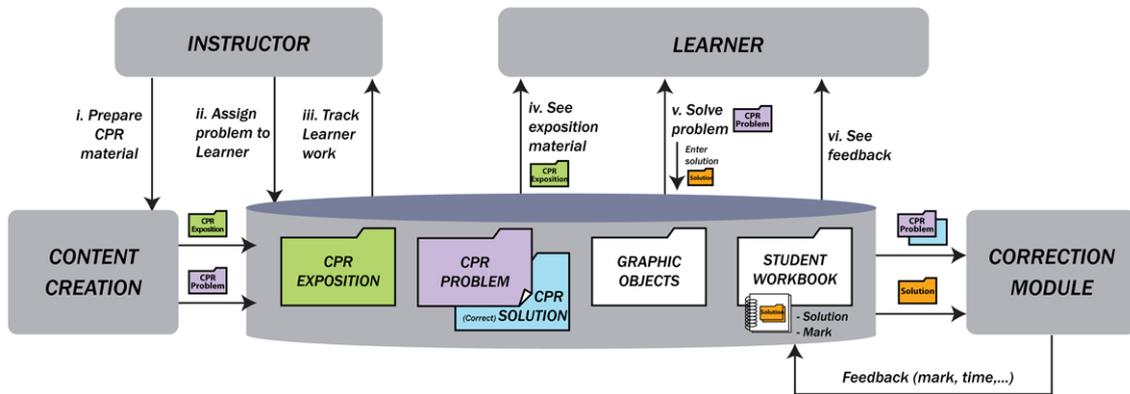


Figure 1: Main modules of LISSA framework

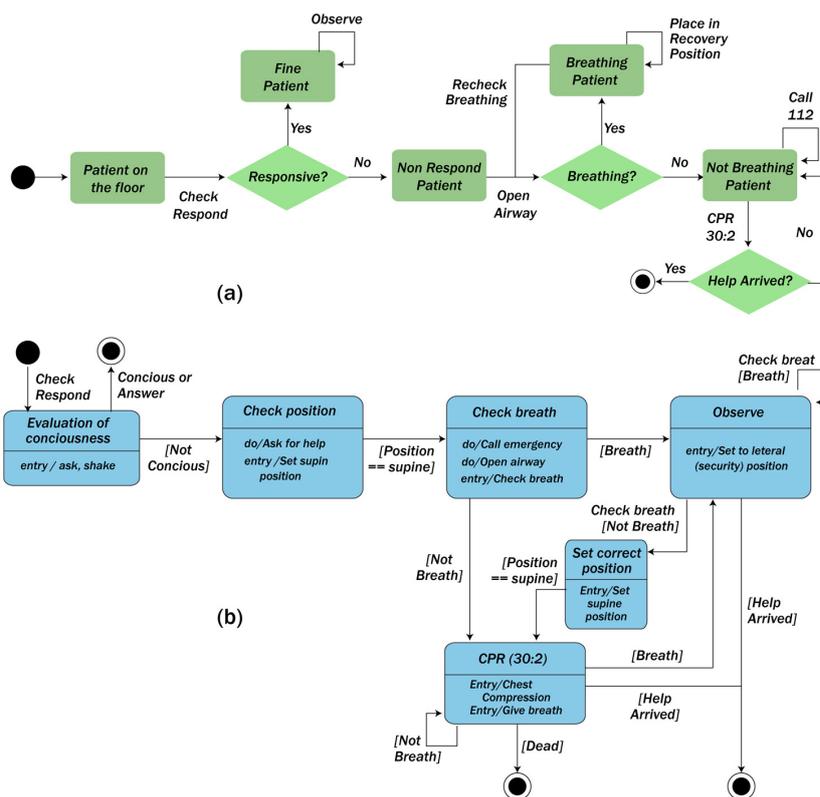


Figure 2: (a) CPR Flowchart for an adult patient [26] and (b) Proposed UML state machine to model the CPR flowchart

To present the functionalities we distinguish between instructors and learners. When the user is an instructor LISSA provides functionalities to: (i) *Prepare CPR material by accessing the content creation module*. The system differentiates between exposition material and problems. Exposition material shows how to apply CPR, using a link to a document or a video. Problems are designed to practice CPR, they are presented as an emergency case. The instructor can modify some of the problem parameters such as the scenario, the victim or the helper. (ii) *Assign problems to the learners*. The instructor can determine the problems and the material he/she wants to assign to the learner. Each learner has assigned a workbook which contains a link to these problems. (iii) *Track learner work*. Since the learner workbook also maintains the entered solutions, the instructor can consult the learner progress.

When the user is a learner, LISSA provides functionalities to: (iv) *See exposition material*. The learner can visualize the content selected by the instructor as many times as he/she wants. (v) *Solve problems*. The system reproduces a CPR emergency situation and the learner applies the CPR procedures which are considered as the problem solution. The system evaluates the solution and assigns a final score (see subsection 3.3). (vi) *See feedback*. For each solution the system creates a report with all the information related to the proposed solution, the learner can consult it to follow his progress.

Two of the main modules of our framework are the content creation, which reproduces the CPR scenario, and the correction module, which determines the score of the learner. Both are described in detail in next sections.

3.2 The Content Creation Module

The content creation module creates the 3D scene with all the elements (actors, medical devices, etc.) that have to be applied to reproduce the situation where the CPR is required. The learner interacts with this scene, has the goal to recover the victim. To create the scene we have to determine how to model the CPR scenario, how to graphically reproduce the scene and how to define the interaction between the learner and the scene.

3.2.1 Modeling the CPR scenario

To model the CPR scenario we used a UML state machine which consists of a set of states and a collection of transitions, which represent some kind of actions for each state. Its purpose is to describe how an object can change its state over time in response to the environment and events that occur. The use of UML state machines in videogames is promoted by many developers due to their robust nature as they are easy to test and modify [11].

Using UML state machine requires a prior planning of all game states and actions. In our case, this information is known and very well-defined by the CPR flowchart provided by the European Resuscitation Council [26]. To illustrate the relationship between the CPR flowchart and the UML-state machine, we present both in Figure 2(a) and (b), respectively. For the sake of simplicity we show a high-level design of the UML diagram. To create the applicable UML state machine, more details need to be completed. For ex-

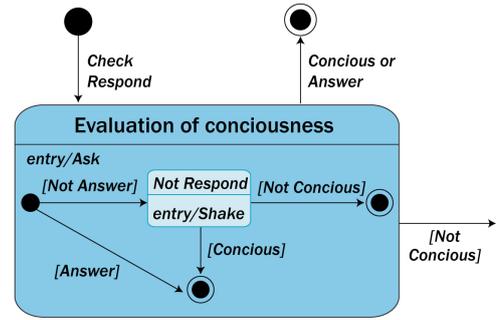


Figure 3: Example of UML state machine (initial stage)

ample, the first state of this UML state machine which represents the state of *consciousness evaluation* corresponds to the diagram shown in Figure 3.

A main advantage of the UML state machine concept is that different scenarios can be easily created by simply modifying parameters such as where the emergency situation is occurred, who is the victim, position of the patient and success percentage. These parameters will be used to determine the level of difficulty of the problems as we will see in Section 3.3.

3.2.2 CPR scene reproduction

Once the UML state machine has been defined, we have to create the scenes and the animations corresponding to the states and the actions (or transitions), respectively. We have to create a scenario-based structure to reproduce a CPR emergency situation where learners will apply their CPR knowledge and skills to recover the victim from the presented situation. Our aim is to reproduce realistic scenes, and for this reason 3D characters and 3D scenes have been chosen.

To ensure real-time interaction and preserve image realism we create scenes where the number of polygons for one real-time render scene, including characters, equipments and environment, are between 40,000 - 200,000 polygons. In addition, most of the texture images have a *jpg* format which can be lossy compressed. On the other hand to represent the transparent objects we use *tif* format since it supports alpha channel. Finally, to illuminate the scenes and the objects we use the lighting functionalities of Unity3D game engine [27].

To create a new problem the instructor has to select from the database the elements of the scene. To support the creation of different CPR scenarios we maintain in a central database models of the possible 3D objects. Figure 4 presents some of these objects. We also allow the instructor to modify some of the parameters related to the objects, such as the position of the victim (lateral or supine), the gender of the patient (male or female) and the location. These parameters can also be randomly determined. Our adaptive system is not useful only for the instructor, in term of providing variety of difficulty problems, but also advantage for an experienced player who want to repeat the use on our game, in term of competence in different situations.

3.2.3 User Interaction

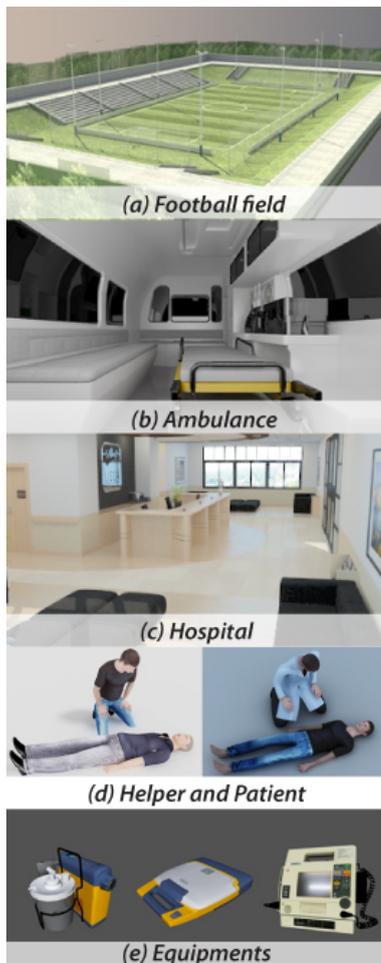


Figure 4: Some of the models stored in the LISSA database and used to create CPR scenes: (a) football field, (b) ambulance, (c) hospital, (d) helper and victim models and (e) some of the supported devices.

Different strategies can be applied to interact with our CPR virtual world [4, 20]. During the graphic user interface design phase of LISSA, we considered two types of realism. The first one is *visual realism* which consists in creating the virtual scene producing the same visual response as the real scene, such as human proportion, character movement, camera viewpoint, etc. The second one is *functional realism* which consists in creating a virtual scene that transmits the same information as the real scene.

Since this first version of LISSA is for PC platform, all the functions start with mouse and keyboard basis. According to the concept of functional realism, we apply two ways to interact with the patient. The first one is based on common widgets such as button, checkbox, drop-down list, text box, etc. The second one aims at simulating the direct interaction with the patient, in this case we apply a mouse-over technique which refers to an event graphic user interface that is raised when the user moves or hovers the pointer over a particular area of the GUI. To identify the parts of the patient's body where the user can interact, we create glyphs

over the part of the patient's body. When learner clicks this glyph, the available options according to that part will appear. In Figure 5 we show a victim with these glyphs. This technique improves human-computer interaction by emulating real-world interactions and providing better ease of use for non-technical people as well.



Figure 5: Glyphs over the patient to show the user interaction areas

3.3 Correction Module

The correction module assigns the score to the learner according to the actions he/she has applied in the CPR situation. When a new problem is created, the instructor has to select a set of parameters that determines the correctness of the proposed solution. These parameters are related to some states of the UML state machine (see Section 3.2.1). For instance, the instructor can select if patient will breath after open airway and the percentage of success after CPR maneuvers. The instructor can also decide the number of attempts that learner can perform to solve the problem.

In addition, the problems can be classified according to their difficulty as easy, normal and difficult. This classification is used to determine the time that will be given to the learner to solve the problem. The more difficult the level is, the less time is allowed to solve the problem. To assign the time for each problem category, we have analysed the time required to solve the different steps of the CPR flowchart and also the different ending possibilities (help arrives, patient recovers or patient dies). After performing different tests we have obtained the average time for each problem category. These times can be assigned by default or can be modified by the instructor. In this way, we allow the learner to perform as many times as he/she wants to solve a problem and with no time restrictions to work in a practice mode.

When the learner enters into the system to solve a problem, he/she has to apply the CPR procedures. The problem will be finished when one of the following situations is given: (i) help arrives, (ii) patient recovers or (iii) patient dies. There is also the possibility to end the problem because time has been expired. In this case, the mission of recovering the patient has not been achieved and the problem restarts. Different performance of solving the problem leads to different scenario ending. We assign a score and time allowed to each one of the actions in the UML state machine. Each time that the learner performs the correct action he/she will achieve the score corresponding to the amount of time used. Otherwise, if the learner performs an incorrect one, the system will not allow him to pass the present state until the correct action will be done. If the learner takes more time than



Figure 6: Graphic user interface of instructor



Figure 7: Graphic user interface of learner

assigned in a transition, it will cause the problem to finish and patient to die. The final score is the sum up of all achieved scores during the process. If the problem is solved with less time than the assigned one the learner will receive extra bonus. In this way we reward correctness and speed.

During problem execution all the learners actions are recorded in the system database and hence we can create a report for each one of the students. This report is presented when the problem finishes and the learner can see his/her progress. This information can also be consulted by the instructor.

4. RESULTS

In this section we present some of the interfaces of the LISSA serious game. LISSA has two main user interfaces, one for the instructor and the other one for the learner. To make the user's interaction as simple and efficient as possible, the user centric design was used. User requirements are considered right from the beginning and included into the whole design cycle. The standard style of interaction such as buttons, list-boxes, checkboxes, textboxes, tabs and scrollbars are used in the graphic user interface components of LISSA similarly to other computational games.

The instructor interface mainly allows instructor to manage the problems and tutorials. Figure 6 shows screen-shots of instructor interface. Graphic user interface of instructor interface is composed of four menus that are exercise, tutorial, student and report. The exercise menu allows instruc-

tor to manage problems by adding, editing, removing and assigning them to learners, see Figure 6(a). To create a new problem scenario, as detailed in Section 3.3, the system authorizes instructor to modify problem parameters and then assign tasks to the learners or to review learner performance and scores later, see Figure 6(b). Next is tutorial menu, see Figure 6(c), similar to exercise menu, instructor is authorized to add the tutorial materials which can be text, image and video

le then assign them to the learners. Student menu presents the list of learners and the classroom in order to view their past performance and assign them the problems. The last menu is report. In this menu, the conclusion of past performance is reported by graphs.

The learner's interface (student workbook) mainly allows learner to solve the problems in the simulation, review an exposition case (tutorial) and review the past performance (history). Figure 7 shows some of the learner interfaces. Figure 7(a) corresponds to the home interface which presents the list of problems assigned to the learner by the instructor. In this list there is also information about the number of previous attempts to solve the problem, time spent in the problem and links to the comments introduced by the teacher. To solve a problem the learner has to select it from the list. Then, the description of the emergency situation appears and the problem starts. Figure 7(b) shows one scene of a problem. All the user actions will be recorded and at

the end a final report will be displayed, see Figure 7(c).



Figure 8: Problem solving

The scene always starts with the emergency situation, when a patient falls down for different reasons in various places, such as football field, swimming pool, bus stop, in the hospital, etc. The learner performs as a helper trying to revive the patient in the limited time. After finishing each mission the score will appear and the learner can choose the next mission for progress. The resuscitation includes a drowning victim, heart attack or stroke patient, or any scenario where breathing or heartbeat have been compromised. In some cases the Automated External Defibrillation (AED) is available.

Figure 8 shows the sequence of actions for a CPR solution from (a) initial state with patient on the floor where learner has to apply the action by clicking on the check respond button then (b) the responsive check is performed, then (c,d,e), which are emergency call, open airway and perform CPR (30:2), 30 chest compressions and 2 breath giving, are applied respectively.

The LISSA project follows a client-server architecture, where the client (main application) requests all the needed information to the server. This information is related to the user, problems and activities. The client part has been implemented using Unity 3D. This software is a game engine that shows the user interface: problem definitions and virtual representations of them (patient, helper and the environment). Unity 3D allows the programmer to write the

code in C#, Javascript or BooScript, but the most used and the one chosen for the development of LISSA is C#. The second part (server part) is implemented using an Apache client. Apache will process all requests that it receives from the Unity3D client and return a response with all the data requested. This information returned to the client is stored in a MySQL database which is installed also in the server, so Apache will have local access to it.

The initial evaluation has been done by experts from faculty of infirmary of our university. Results of the assessment are positive regarding to the user interface, realism and resources management (problems, learner's performance). Further studies about score strategy, learning step, advanced procedure of basic life support are needed to improve the application.

5. CONCLUSION

Cardiopulmonary respiration (CPR) is a crucial part of basic life support and different strategies have been proposed to teach the CPR procedures. In this paper we have proposed a new strategy based on videogame technologies. Our LISSA proposal is a serious game for teaching and learning CPR. The game turns around an emergency case which has been modelled as a UML state machine that reproduces a CPR flowchart. 3D graphical representations and animations have been proposed to model the states and the transitions between the states. LISSA supports two type of users, instructors and learners, and integrates different e-learning functionalities to allow the definition of problems of different levels of difficulty, the assessment of the learners or the consultation of actions performed by learners. The first version of LISSA, based on PC with mouse and keyboard, has been evaluated positively by a group of experts from the nurse faculty of our university. As a future work we plan to integrate Kinect into platform for better real-like interaction. Moreover, we will do an exhaustive evaluation of the platform in different groups of learners. We also plan to generalize this environment to deal with any kind of life support procedure and integrate new interaction techniques to enhance realism.

6. REFERENCES

- [1] Insight Instructional Media LLC. AED Challenge. "http://aedchallenge.com/", 2011. Accessed 15 Dec 2012.
- [2] American Heart Association. Emergency Cardiovascular Care online education. "http://www.onlineaha.org", 2012. Accessed 30 May 2012.
- [3] M. Boyer. First Aid & CPR Basics, First Aid for Choking Victims. "http://www.youtube.com/watch?v=htzRcv1Z6_s&feature=relmfu", 2007. Accessed 29 May 2012.
- [4] F. Bruno and M. Muzzupappa. Product interface design: A participatory approach based on virtual reality. *International Journal of Human-Computer Studies*, 68(5):254 – 269, 2010.
- [5] College to become an EMT. How to Perform CPR. "http://www.youtube.com/watch?v=cPEFskCrhdhQ", 2008. Accessed 29 May 2012.
- [6] D-Sign S.r.l. iCPR - iPhone App for CPR training.

- "<http://www.icpr.it/>", 2009. Accessed 19 Dec 2012.
- [7] Dassault Systèmes - Ilumens - Medusims. Staying Alive 3D. "http://www.stayingalive.fr/index_uk.html", 2011. Accessed 15 Dec 2012.
- [8] EM Gladiators LLC. CPR Game. "<http://www.freenew.net/iphone/cpr-game-20/466678.htm>", 2011. Accessed 15 Dec 2012.
- [9] FirstAidWeb Inc. Learn CPR. For Life. "<http://www.firstaidweb.com>", 2012. Accessed 30 May 2012.
- [10] J. M. Fisher. ABC of resuscitation. Recognising a cardiac arrest and providing basic life support. *British Medical Journal (Clinical Research Ed.)*, April 12:1002-4, 1986.
- [11] W. Hu, Q. Zhang, and Y. Mao. Component based hierarchical state machine; a reusable and flexible game ai technology. In *Information Technology and Artificial Intelligence Conference (ITAIC), 2011 6th IEEE Joint International*, volume 2, pages 319-324, aug. 2011.
- [12] Indiegogo Inc. Mini-VREM. "<http://www.indiegogo.com/mini-vrem>", 2012. Accessed 15 Dec 2012.
- [13] W. B. Kouwenhoven, J. R. Jude, and G. Knickerbocker. Closed chest cardiac massage. *Journal of the American Medical Association (JAMA)*, pages 84-97, 1960.
- [14] Laerdal Medical. MicroSim Prehospital. "<http://www.laerdal.com/gb/docid/12984879/MicroSim-Prehospital>", 2012. Accessed 15 Dec 2012.
- [15] Less Stress Instructional Services. Adult CPR Simulator 2.0. "http://www.qweas.com/download/home_education/health_nutrition/adult_cpr_simulator.htm", 2012. Accessed 31 May 2012.
- [16] Low D. and Clark N. and Soar J. and Padkin A. and Stoneham A. and Perkins G. and Nolan J. Does use of iresus application on a smart phone improve the performance of an advanced life support provider in a simulated emergency? *Resuscitation*, 81(2, Supplement):S10 -, 2010.
- [17] Medtronic Foundation. CPR Simulator. "<http://www.cprsim.com/>", 2012. Accessed 31 May 2012.
- [18] Medtronic Foundation. Save-A-Life Simulator. "<http://www.heartrescuenow.com>", 2012. Accessed 15 Dec 2012.
- [19] National Safety Training. online safety training. "http://www.nsc.org/products_training/Training/Pages/OnlineTraining.aspx", 2012. Accessed 30 May 2012.
- [20] Z. Pan, A. D. Cheok, H. Yang, J. Zhu, and J. Shi. Virtual reality and mixed reality for virtual learning environments. *Computers & Graphics*, 30(1):20 - 28, 2006.
- [21] Ponder, M., Herbelin, B., Molet, T., Scherteneib, S., Ulicny, B., Papagiannakis, G., Magnenat-Thalmann, N., Thalmann, D. Interactive scenario immersion: Health emergency decision training in just project. *Proc. Of 1st International Workshop on Virtual Reality Rehabilitation, VRMHR2002, VRLab-EPF*, pages 87 - 101, November 2002.
- [22] RedCrossOnlineTraining.org. Red Cross online training. "<http://redcrossonlinetraining.org/Distance/Default.aspx>", 2012. Accessed 30 May 2012.
- [23] F. Semeraro, A. Frisoli, C. Loconsole, F. BannÃš, G. Tammaro, G. Imbriaco, L. Marchetti, and E. L. Cerchiari. Motion detection technology as a tool for cardiopulmonary resuscitation (cpr) quality training: a randomised crossover mannequin pilot study. *Resuscitation*, (0):-, 2012.
- [24] F. Semeraro, F. Taggi, G. Tammaro, G. Imbriaco, L. Marchetti, and E. L. Cerchiari. icpr: A new application of high-quality cardiopulmonary resuscitation training. *Resuscitation*, 82(4):436 - 441, 2011.
- [25] Stone Meadow Development LLC. CPR & Choking. "<https://itunes.apple.com/us/app/cpr-choking/id314907949?mt=8>", 2011. Accessed 15 Dec 2012.
- [26] The European Resuscitation Council. ERC Guidelines 2010. "<http://www.cprguidelines.eu/2010/>", 2012. Accessed 30 May 2012.
- [27] Unity Technologies. Create the games you love with Unity. "<http://unity3d.com/unity/>", 2012. Accessed 15 January 2012.
- [28] R. Zanner, D. Wilhelm, H. Feussner, and G. Schneider. Evaluation of m-aidÃš, a first aid application for mobile phones. *Resuscitation*, 74(3):487 - 494, 2007.