

Development of an Educational Spatial Game using an Augmented Reality Authoring Tool

Claudio Kirner¹ and Tereza G. Kirner²

Department of Mathematics and Computer Science, Federal University of Itajubá, Brazil

¹ ckirner@unifei.edu.br, ² tgkirner@unifei.edu.br

Abstract: Developing an augmented reality application is usually a task for expert programmers, but using appropriate resources a non-programmer can be successful in such intent. This paper discusses the development of an educational spatial game to be customized by non-programmers users and shows the used authoring tool, the development process of the game and its usability evaluation considering the augmented reality characteristics. Finally, we discuss the game authoring and the customization approach and point out directions to improve the tool and the game.

Keywords: Spatial game, Augmented reality, Educational application, Authoring tool, Customization, Evaluation.

I. Introduction

Spatial applications mediated by computer including navigation, points of interest search, three-dimensional games, three-dimensional educational tools, etc., are growing with the use of augmented reality (AR).

Nowadays, the technological convergence allows the development of applications involving many resources working together, such as GPS, accelerometers, cameras, mobile phones, notebooks, etc. The creation of those applications depends strongly on programmers and people with technical knowledges.

However, there are many non-programmers people interested to develop simple and complex spatial applications based on augmented reality. To overcome this problem, some augmented reality tools are providing resources to allow non-programmers to generate their own augmented reality applications [17]. It is important to mention that these applications must be independent of compilers, focusing text edition, configuration tasks, visual interface or tangible actions.

Therefore, the main question in this paper is: how can non-programmers develop their own AR applications by editing and configuring textual parameters and spatial elements? To answer this question, we will present firstly an analysis of AR authoring tools emphasizing their authoring features that are non-dependent on programming or technical skills. Then, we will present an augmented reality authoring system for non-programmers (ARAS-NP) developed to make easy the authoring process by non-expert users. After that, we will present an educational application developed with the tool

ARAS-NP and its evaluation from the point of view of augmented reality. Finally, we present the conclusions of this work.

II. Related work

During the last ten years, several AR tools were released to build spatial applications mediated by computer.

Authoring augmented reality tools can be classified according to their characteristics of programming and content design in low and high levels, presenting different concept abstraction and interface complexity [5], [17]. Programming is usually more complex, has more flexibility and presents less concept abstraction than content design. Content design tends to be less complex, less flexible and presents more concept abstraction. Therefore, as the concept abstraction grows, the interface becomes less complex and less flexible.

Programming tools are based on basic or advanced APIs containing libraries involving tasks like computer vision, registration, three-dimensional rendering, sounds, input/output and other functions. ARToolKit [7], MR [18] and MX [3] are examples of low level programming tools, while Studierstube [16], OsgART [14] and DWARF [1] comprise high level programming tools.

ARToolKit is one of the first augmented reality tools that use marker registration and computer vision. The developers need C/C++ programming skills to author augmented reality applications.

Content design tools remove dependency on a programming language, replacing it by the description of the virtual objects and their relationship with the real environment. APRIL [13] is a lower level example of this type of tool, depending on XML descriptions. High level content design tools use graphical user interfaces to represent the descriptions and interactions, as are the cases of DART [15], AMIRE [4], ECT [5], ComposAR [17] and ARSFG [2].

DART is a tool implemented over the Macromedia Director using a drag-and-drop authoring model and an interpreted scripting language.

AMIRE is a framework that uses a component oriented technology and presents a minimal set of components for demonstration, reusable resources collection and a visual authoring tool.

ECT is a component oriented software toolkit with a visual

programming interface designed to ease the development of AR applications by users with little or no computer programming experience.

ComposAR is an extensible authoring tool for AR authoring by non-programming users. It supports scripting and drag-and-drop interface, real time interpreted input and functionality added by users.

ARSGF is a software framework for rapid AR prototyping based on scene graph that allows remote collaboration through a XML based protocol.

High level content design should be more intuitive and suitable for non-programmers. These tools can support scripting and/or visual interface, new functionalities added by user and real time interpretation.

Our research is different from those AR interfaces since it uses the following characteristics together:

- We use an authoring tool implemented over ARToolKit;
- Authoring AR applications depends on editing configuration files and/or tangible operations;
- There are different authoring levels depending on the skills of the developer who does not need to be a programmer;
- There is a collaborative support to allow multiple remote users working on a shared AR environment using tangible operations;
- Authoring can use tangible operations, editing configuration files and mouse and keyboard support, but the end users can interact with the AR application using only one or two markers.

III. Authoring tool to develop augmented reality applications

Authoring AR applications depends on: structure of the AR environment; data structure and folders that support the tool; authoring interface, including configuration tasks, action commands, system commands; and utilization procedures that support end-user to navigate and interact with the augmented environment.

To make easy the development of AR applications with those elements, we developed the authoring tool ARAS-NP [9]. It has authoring and using characteristics, explained in this section, besides additional features related to shared remote use for collaboration.

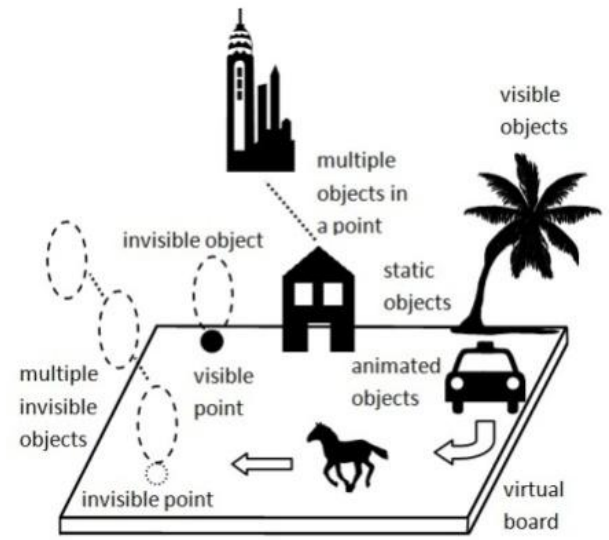
ARToolKit is the core of ARAS-NP and additional functionalities were programmed with C/C++. The software, user manual and applications of ARAS-NP are freely distributed by the authors [11].

A. Structure of the augmented reality environment

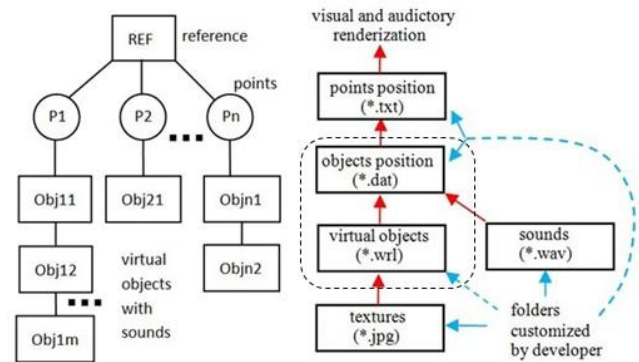
Augmented reality involves more than superimpose virtual objects and annotations over the real world. Thus, the augmented world (Figure 1a), as considered in this work, presents real and virtual objects, such as interactive objects, which can change in certain situations; animated objects, which can be activated; visible or invisible objects, which vanish or appear in certain cases; visible or invisible points, which can be activated or deactivated; etc.

Moreover, the augmented reality environment can be modified after the initial authoring by creation, changing and

deleting points and virtual objects.



a) Augmented world



b) Data structure

c) Folders

Figure 1. Structure of the augmented reality environment

B. Data structure and folders

The data structure of the augmented reality environment to be authored comprises reference markers, which have associated virtual boards, and their respective elements (points, virtual objects and sounds) that appear on the board, according to Figure 1b. These elements must be placed in folders (Figure 1c) that the developer needs to manipulate in order to create the augmented reality environment.

The tool has five types of folders configurable by developer: points and objects position, virtual objects, sounds and textures. Details about folders will be explained in the next subsection.

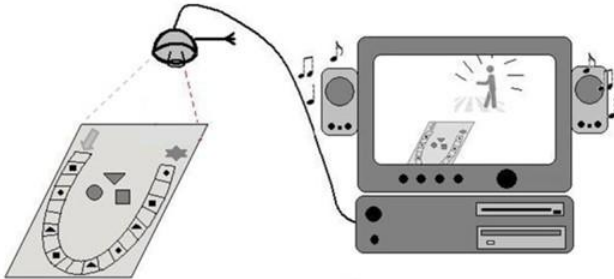
C. Authoring interfaces

In the authoring phase, the application can be developed by visual actions carried out under a webcam (Figure 2a) and/or editing files. The resources, which are used to create points and associated objects with sounds, comprise action markers, keyboard keys, mouse buttons, configuration files and folders.

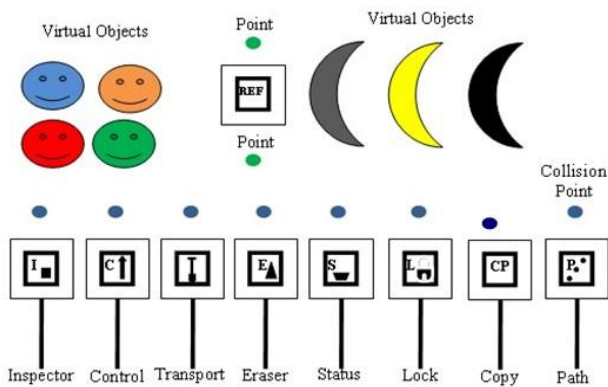
1) Action Markers

Action markers allow tangible actions manipulating points and virtual objects placed on or over the respective virtual board, according to Figure 2b.

In this environment, points and virtual objects have visible or invisible virtual spheres associated with them, whose radius can be adjusted by user, allowing variation in precision during the selection process. Each action marker also has an associated visible virtual sphere that can collide with the virtual spheres of points and virtual objects.



a) Augmented reality environment



b) System markers

Figure 2. Augmented reality environment and its markers

An action marker can select a point or virtual object and carry out a specific function on it. The movement of an action marker in the board area allows collision of its virtual sphere with the virtual sphere of a point or virtual object of the augmented environment, selecting or deselecting the point or virtual object. The gesture of the action marker, like inclination, occlusion, or time remained selecting, can execute actions on the selected point or virtual object.

These markers are: inspect (showing the virtual object of the selected point); control (showing the next virtual object of a list associated with the point); transport (taking a virtual object from a position to another one); erase (deleting a point and its associated virtual object or list); status (showing a virtual plate with information about the system); lock (locking or unlocking remote actions on virtual objects associated with a point, being useful in collaboration tasks); copy (replicating a virtual object to be placed in a position of other reference environment), and path (allowing the creation and visualization of a visual path from a position to another one).

2) Keyboard Keys and Mouse Buttons

Specific keyboard keys and mouse buttons activate complementary actions working alone or in conjunction with action markers.

Those keys and buttons can activate or deactivate: visualization of points and virtual objects, remote operations, path creation, status visualization, persistence of virtual

objects, etc. They can also control the position and radius of the virtual sphere associated with the marker, aiming to improve precision and performance in the selection process. As less is the radius of the virtual sphere, more difficult is the selection, but in some cases, when there are many closed points, it is necessary to get better precision to be successful in the operation.

Mouse buttons actuating in conjunction with the inspector marker allow visual creation of points in the augmented environment and in the respective file that can be configured later by editing.

3) Configuration Files

Configuration files must be created by the user or edited from the sample files available in the authoring system or from previously created files in the visually points creation process.

These files contain authoring information supplied by the user in order to prepare the augmented reality environment for use or for additional visual configuration. The configuration files are:

1. Reference Marker Information (one file for each reference marker):
 - indication of the virtual board associated with the marker;
 - position, orientation and scale of the virtual object;
 - indication of the sound associated with the virtual board, if it exists;
 - indication of the points list.
2. The Points List of a Reference Marker (one file for each reference marker):
 - position and list of virtual objects associated with the point 1;
 - position and list of virtual objects associated with the point n.
3. The Virtual Objects List of a Point (one file for each point):
 - virtual object 1:
 - indication of the three-dimensional model of the virtual object 1;
 - position, orientation and scale of the virtual object 1;
 - indication of the sound associated with the virtual object 1;
 - virtual object m: this information is similar to the virtual object 1.

A simple text editor should be used to create or edit configuration files.

4) Folders

Folders are resources filled by non-programmer developer, which contain useful information to create the augmented reality environment. There are four folders in the system that are used by developers: configuration information (points and objects position), virtual objects, sounds and textures, according to Figure 1c.

Configuration information folders contain configuration files related to reference markers, points and virtual objects. Virtual objects folder contains VRML three-dimensional

models, found on the Internet or created by three-dimensional authoring software. Sounds folder contains sounds associated to virtual objects found in sounds libraries or recorded by the developer. Textures folder contains images used by virtual objects to be more realistic or rendering information.

Folders must be filled with content before using, but they can be modified during the utilization phase.

The system also has commands to save and recover an authoring process aiming to allow the interruption of development and its restoration later.

D. Utilization procedures

In the utilization phase, the end-user should manipulate the augmented reality environment using tangible actions with specific markers. The most used are the inspector and control markers, which allow the exploration of the environment, showing what is on each point by activating virtual objects and its related sounds from the list associated with it.

However, aiming to give more power to the system, it is possible to reconfigure the environment during the utilization phase, changing the initial situation of the authoring. This procedure provides a static or dynamic customization of the authoring, depending on the time or periodicity of the changes. In this case, the user can: change the visibility of points and virtual objects; exchange the current virtual object by another one from the list associated with the point; make copies of virtual objects; delete points and virtual objects. These actions are simple and require less effort in comparison to the initial authoring.

Therefore, at any time, the user can reconfigure the augmented reality environment making it unforeseen or personalized. This characteristic is useful in the development of AR games and educational applications.

Moreover, the system has a specific marker (collaborative ref marker), which allows the networking among several remote users, implementing collaborative tasks. When each user put the collaborative ref marker in his/her environment, he/she is connected to the network in order to see the shared augmented reality environment so that he/she can see and provide actions to change it. This feature is useful in games and educational applications, allowing collaboration.

IV. Development of an application with the ARAS-NP

Many augmented reality applications were developed with the authoring tool ARAS-NP, such as interactive augmented reality book, educational games, etc. We choose the development of an educational spatial game to illustrate the use of the tool in the authoring process.

A. Educational game based on questions and answers

The educational spatial game, named Q&A-AR, is a multiplayer car racing game based on questions and answers that works in an augmented reality environment.

The game Q&A-AR must fulfill the following requirements:

- The game must have educational potential involving several themes of study using texts, illustrations and sounds.
- The physical parts of the game must be made with ordinary materials and process in order to have availability and low cost.

- The logical parts of the game must use augmented reality technology based on computer vision.
- The interactive actions to be executed on the game must be tangible and easy.
- The information related to the game (questions, answers, instructions) must be easily customized by teachers.
- The user interface of the game must consider usability factors, such as easy to understand, easy to learn and easy to use.

B. Implementation of the game structure

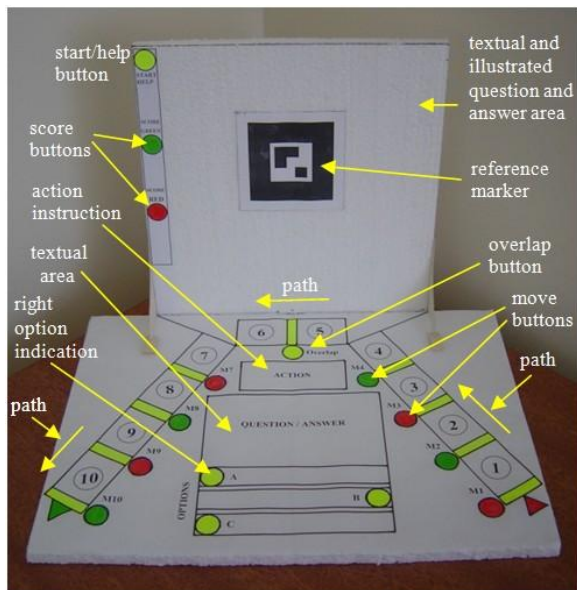
In order to implement the game satisfying those requirements, we made the following decisions:

- Using a three-dimensional structure that allows the implementation of educational questions and answers, involving space for texts, illustrations and instructions for the game functioning. We choose a structure with two planes (vertical and horizontal), containing the displaying area and several printed buttons to be used as a guide for user interaction. We also choose to render information on virtual planes overlapping the physical ones and showing virtual buttons over the printed ones.
- Using Styrofoam (FPS), cardboard or wood to implement the physical structure tied with glue, following instructions and templates that will be available on the Internet.
- Using the augmented reality authoring tool ARAS-NP for rapid prototyping and customization.
- Using only two physical pointers with coupled markers in order to make the interaction easy.
- Implementing textual, auditory and illustrated questions and answers to be attractive to the user.
- Considering usability and other factors applied to augmented reality user interface, such as colored buttons, information displayed on layers in order to reuse the same displaying area, sound and narration to complement the virtual information, overlapped physical and virtual buttons to make the interaction easy and fast.

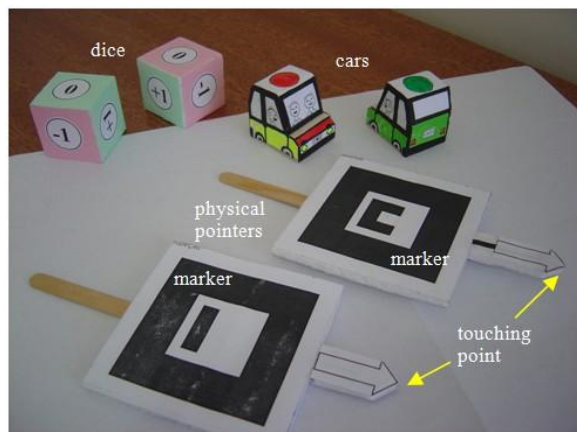
The game has two structures to implement the augmented reality environment: the physical structure and the virtual structure.

The physical structure has two types of elements composed by nonmoving planes with printed information and moving elements.

The physical nonmoving structure (Figure 3a) contains two perpendicular planes (vertical and horizontal) in order to present the game information to the user. The vertical plane contains the reference marker, which is used to superimpose the virtual information on physical elements, and three control buttons (start/help button and two score buttons for two individual players or two teams). Besides, the vertical plane will receive textual and illustrated questions and answers. The horizontal plane presents the race path with ten cells, a textual area for questions and answers, an area for action instructions, buttons for indication of forward and backward car movements, a button for decision about car overlapping on a path cell and three buttons below the questions and answers area to indicate the correct answer option.



a) Nonmoving structure

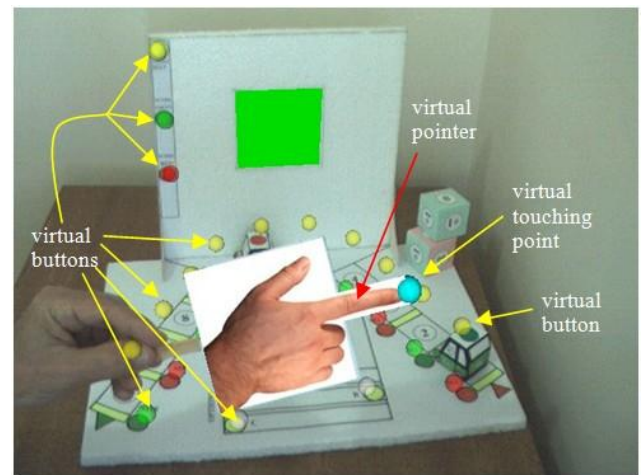
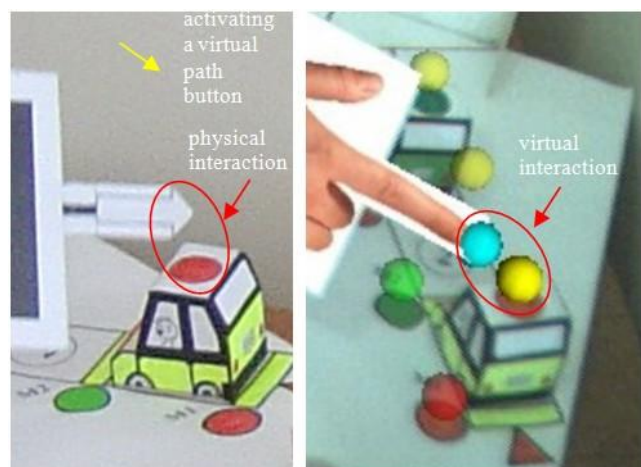


b) Moving parts

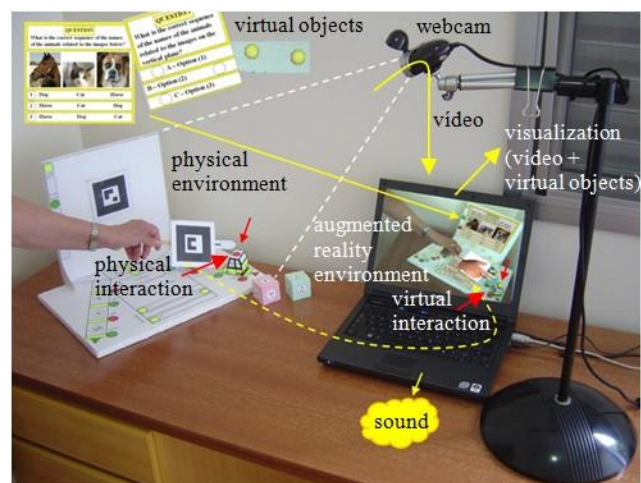
Figure 3. Physical structure of the game

The physical moving parts (Figure 3b) are composed by two colored cars (red and green), two colored dice and two interaction pointers with markers. The two dice have red faces with zero or negative numbers (for backward car movements) and green faces with zero or positive numbers (for forward car movements). One die has five red faces and one green face and the other has five green faces and one red face. The two physical interaction pointers present different markers: one, which has the symbol "I", is used to activate and deactivate virtual buttons in order to replay visual and auditory information and the other with the symbol "C" is used to render the next information (visual and auditory) from a list associated with each virtual button.

The virtual structure, according to the Figure 4, is composed by virtual buttons overlapping the printed buttons, virtual path cell buttons, which will appear over the physical cars when they are placed on path cells, and a virtual pointer with a virtual touch point used to interact with the virtual buttons. To make the interaction easy and fast, the player only needs to touch the physical pointer on the printed buttons or on the top of the cars placed on the path cells (Figure 5). An equivalent virtual interaction will occur in the augmented reality environment that will emit a sound indicating that the interaction was carried on.

**Figure 4.** Virtual elements of the game**Figure 5.** Physical and virtual interactions

The augmented reality environment of the game, according to Figure 6, shows the physical environment captured by the webcam and transformed into a video and the virtual objects (virtual planes, virtual buttons and virtual pointer) that overlap the video on the monitor.

**Figure 6.** Augmented reality environment of the game

The cars run over the path on the horizontal plane driven by moving information, involving dice, forward and backward movement indicated by buttons or by result of the player performance responding the questions presented by the activation of the virtual path cell buttons. The goal of the game

is to reach the end of the path first. One variation on the goal is to reach the end of the path with the higher score involving the players. In this case, if a player does not more points than the other one and reaches the end of the path, he/she will need to continue playing until get enough points to win.

C. The game in action

Before starting the game, the player can see the general instructions touching the yellow button (start/help) on the top-left position of the vertical plane (Figure 7) with the pointer "I". After that, touching the same button with the pointer "C" many times, the player will receive further instructions about the game, page by page. Touching that button with the pointer "I" again, the start/help instructions will be deactivated.

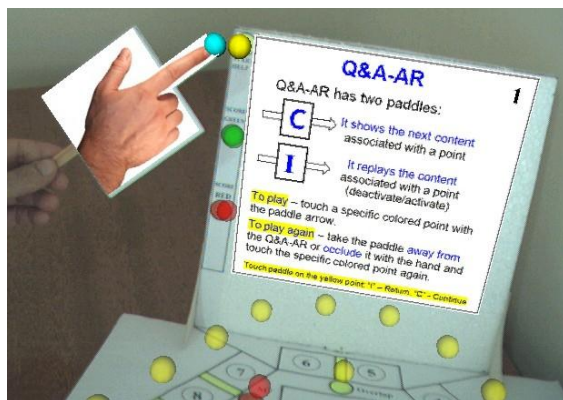


Figure 7. Activating the instructions about the game

In order to prepare the game to work, the player must press the key <a> on the computer keyboard activating all virtual buttons at the same time (all virtual buttons will become colored).



Figure 8. Activating the question

The game starts when the first player places his/her car (with the green or red button printed on top) on the first path cell or throws the die placing the car on the path cell indicate the top cell of the die. After that, the player can touch the button over the car (printed green or red button) with the pointer "C", activating the corresponding virtual yellow button

that will display the first question on the vertical and horizontal planes, emitting the respective sound and/or narration (Figure 8).

After hearing and reading the question, the player must indicate the correct answer option, activating the corresponding button, which will become blue (marked button), according to Figures 9 and 10.

Hence, the player can use the pointer "C" to touch the printed button over the car, interacting with the corresponding virtual yellow button. That operation will replace the question by the answer information on both planes and change the color of the virtual yellow button over the car to blue (Figure 11).



Figure 9. Activating the correct answer option



Figure 10. Activating the answer

In this situation, the player can check the answer whose correct option is marked as a blue row on the vertical plane and as blue coin closed to the option button on the horizontal plane. If the blue coin and the marked button are closed, the user answer is correct otherwise the user answer is wrong. Depending on the answer information provided by the system,

the action area on the horizontal plane will indicate the car movement based on explicit number of path cells that the car must go forward (green field) or backward (red field), or a way to get the car movement information involving throwing an specific die or touching an specific move button.



Figure 11. Showing the answer elements

Before moving the car, the player needs to begin the deactivation of his/her information, preparing the game for the other player (Figure 12).

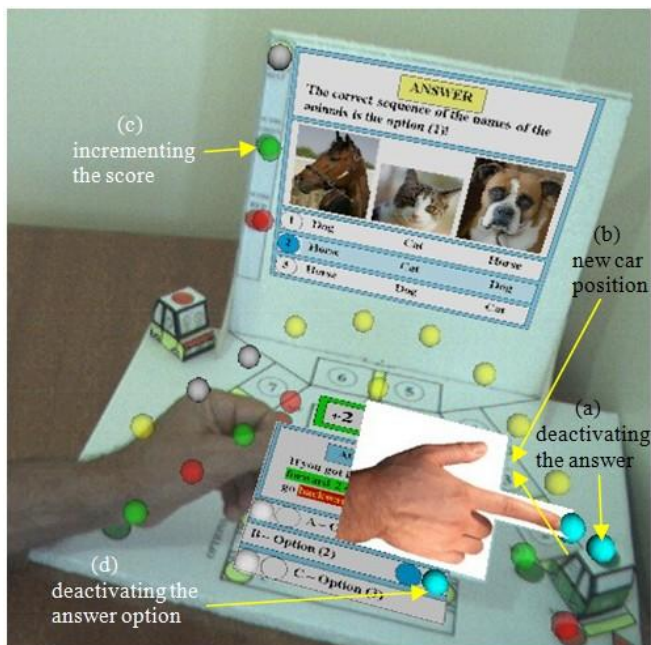


Figure 12. Preparing the game for the other player

The following procedures prepare the game for the next player.

- Touching car button with the pointer “C”, deactivating the answer information on both planes and its corresponding virtual path button that will change from blue to yellow.
- Moving the car to the path cell indicated by the car movement information received during the answer check.

- Touching the corresponding score button (green or red) on the vertical plane with the pointer “C” to increment its value (Figure 13).
- Touching the marked answer option with the pointer “C”, deactivating it. The virtual blue button will become white.
- If a move button is activated, it must be deactivated by the pointer “C”, changing from blue to red or green.

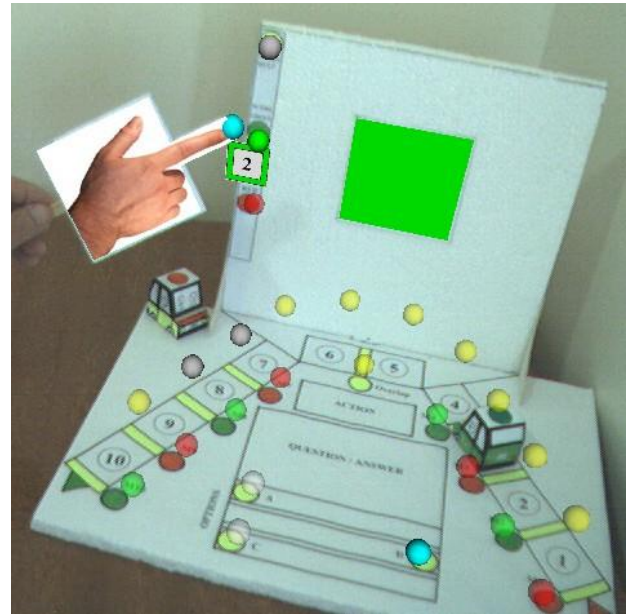


Figure 13. Incrementing the score

In such way, the game is ready to be played by the second player who can place his/her car on an initial position, according to the game rules (Figure 14), or continue the game if it is already running. The player must touch the printed car button with the pointer “C” following the visual and auditory instructions.

Every time the pointer touches a button, it needs to be moved outside the webcam vision field or occluded by a hand to work again. If the player wants to hear the information again, he/she must use the pointer “I” twice in order to deactivate and activate a virtual button rendering the visual and auditory information.

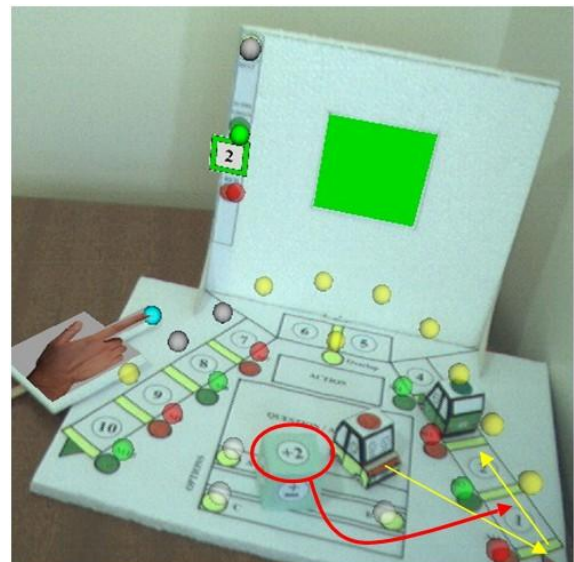


Figure 14. Placing a new car on the game

D. Development of content and customization

To allow the rapid development of content, the game Q&A-AR is based on textures and sounds that can be prepared by teachers.

Each virtual path button has an associated list of repeated sequences composed by an empty content, question and answer. The pointer “C” interacting with a virtual path button allows forward navigation on the elements of the list, returning to the beginning when reaches the end of the list. Each question or answer presents the same structure, but points to different textures and sounds and/or narrations in each case. To change the question and answer, it is enough to prepare new question and answer texture files and record new sound and/or narration file, placing them on the corresponding folders replacing the old files by the new ones with the same file names (Figure 15).

To customize the game Q&A-AR, a teacher must follow the process below.

- Check the texture folders related to questions and answers to see how many textures there are in each one.
- Prepare the same number of illustrated questions, textual questions, illustrated answers and answer actions using a text editor and the templates in each case.
- Capture the screens area of prepared questions, answers and actions as images like jpg, for example, and rename them with the same names of the equivalent files inside the texture folders.
- Replace the old textures by the new ones.
- Record new sounds and/or narrations associated with each question and answer, creating the same file names of the old ones.
- Replace the old sounds and/or narrations by the new ones.

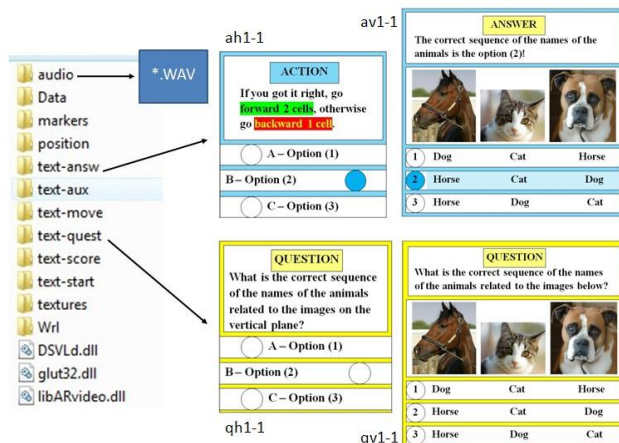


Figure 15. Texture and sound folders and files

If the teacher wants to increment the amount of questions and answers associated with a specific virtual path button, he/she will need to prepare the new textures and sounds/narrations, place them in the respective folders and edit each path(i).dat file contained in the wrl folder, according to Figure 16. Besides, the new questions and answers structures must be created by replication and edited to change its texture links considering the new textures.

```
wrl/path1/ball-y.wrl
0 0 0          # Translation
0.0 0.0 0.0 1.0 # Rotation
10 10 10       # Scale
audio/pause    # audio file
0 0

wrl/path1/Q1-1.wrl
-175 115 -223  # Translation
0.0 0.0 0.0 1.0 # Rotation
10 10 10       # Scale
audio/question1-1 # audio file
0 0

wrl/path1/A1-1.wrl
-175 115 -223  # Translation
0.0 0.0 0.0 1.0 # Rotation
10 10 10       # Scale
audio/answer1-1 # audio file
0 0
```

Figure 16. Content of the file path1.dat

In case of decreasing the amount of questions and answers, it is necessary to edit the path(i).dat file deleting the unwanted blocks.

With this process, the teacher can customize the game in order to explore specific themes and different amounts of questions, answers and actions, improving the learning quality of students.

E. Evaluation of the application

A previous version of the educational game [10] was tested and evaluated in some aspects of usability related to augmented reality [8], [6] and the results were used to improve the current version of the game Q&A-AR. The evaluation was done with 17 computer science undergraduate students who played the game following the visual and auditory instructions and responding the game questions. Before testing, the students had a game demonstration and after testing they filled a questionnaire responding 12 questions about augmented reality aspects of the game (Table 1). For each question, they chose one from five options (5 -Excellent, 4 - Very good, 3 - Satisfactory, 2 - Fair, 1 - Poor), according to Likert scale. The 12 questions were collected into four groups of usability factors evaluating the application from the point of view of interaction, interface, visual aspects and audio aspects. Each group has three items that were analyzed by the users.

Table 1. Augmented reality aspects of the educational spatial game

Usability Factors	Analyzed items
(a) Game Interaction	(1) Marker handling; (2) Marker collision with virtual points and objects; (3) Response time of the interaction.
(b) Game Interface	(4) Easiness, learnability and usability; (5) Attractiveness; (6) Interaction with game resources.
(c) Visual Aspects	(7) User physical support involving markers, guides, cars, etc; (8) User virtual support involving paths, points, texts, etc; (9) Physical and virtual resource integration.
(d) Audio Aspects	(10) Audio involving game instructions; (11) Audio involving questions and answers; (12) Sound levels.

The results of the evaluation can be seen in Figure 17.

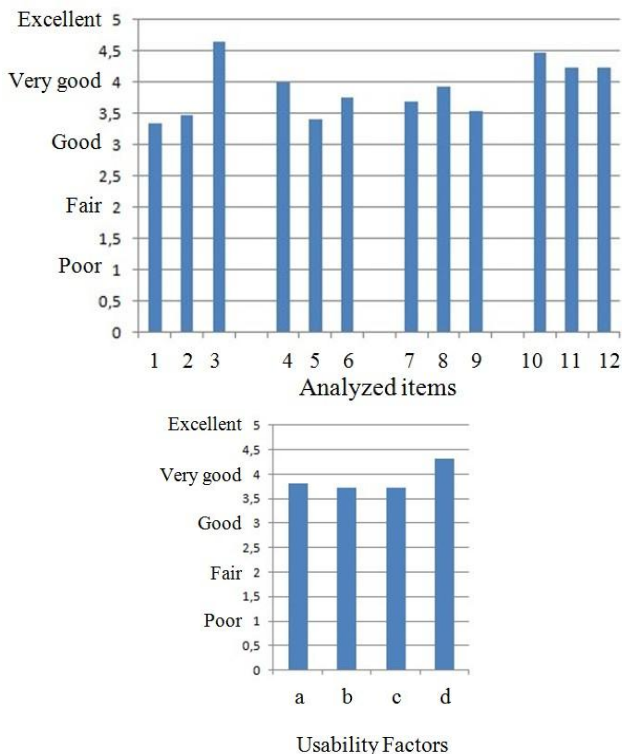


Figure 17. Evaluation of the educational spatial game (see Table 1)

The best results were obtained by items (3) “Response time” and (10) “Audio involving game instructions”, giving a rate between “Very good” and “Excellent”. The worst results came from items (1) “Marker handling” and (5) “Attractiveness”, giving a rate between “Satisfactory” and “Very good”. The average of all items is almost “Very good”.

Considering the usability factors, the best result was found on audio aspects with a rate better than “Very good” and the worst result was observed on interface and visual aspects with rates between “Satisfactory” and “Very good”.

Those aspects of usability related to augmented reality features of the educational spatial game depends on the following factors: a) Environmental factors, such as illumination, webcam quality, marker pattern resolution; b) Quality of the developing tool (ARAS-NP) bringing an heritage of the used core (ARToolKit); c) Talent and developing skills of the developer; d) User skills in using augmented reality technology.

Considering that the developer, as well as the users did not have experience with the tool and the AR technology, since all of them are undergraduate students, and considering uncontrolled environmental conditions in the classroom where the test was carried out, we believe that the results were good. However, it is possible to improve the application with better: game elements (virtual objects, questions and answers, marker patterns), computational resources (computer, webcam), environmental conditions (illumination, positions) and developer/user training.

From the observations, the tool ARAS-NP seems to be appropriated for development of educational spatial applications, once it has several resources for authoring and handling whose developed applications were well evaluated by end users.

V. Conclusion

We presented and discussed the problem of authoring augmented reality applications by non-programmers and described the tool ARAS-NP from the point of view of non-expert authors. We also showed the tool in action during the implementation of a spatial game Q&A-AR with positions, virtual objects, annotations and sounds.

Although the ARAS-NP tool uses different types of authoring resources, such as markers, keyboard keys, mouse buttons, and configuration files, the authoring process can be simple when the designer uses textures and sounds as the main elements of the application.

The author does not need to be expert, but he/she should present some skills like discipline, planning, spatial awareness and basic computer knowledge, such as file editing, sound recording, screen capture, folder manipulation, and eventually three-dimensional modeling.

The development of a spatial game using the tool demonstrates that it is possible to use few resources in authoring and few markers (reference, inspector and control) during the utilization time.

The educational spatial application was evaluated demonstrating that its augmented reality characteristics are good but can be improved. Moreover, the use of ARAS-NP showed that it is a suitable tool to develop spatial applications in a short space of time with good parameters related to interface, interaction, visual aspects and auditory aspects.

Moreover, the spatial game was designed to be easily customized by non-programmers, allowing teachers to adapt it for different interest theme. In the next months, we will release a game pack for download [12], including: templates, software, tutorials, videos, and different educational themes applications.

Finally, we believe that the tool ARAS-NP can be improved to be more attractive to non-programmers in order to be a good option for rapid development of augmented reality applications. The game Q&A-AR can be customized and distributed on the Internet to be used by teachers and students as an attractive educational application.

Acknowledgment

This research was developed as part of the project “Internet Environment to Support Teachers in the Development of Educational Applications with Augmented Reality” funded by Brazilian Agency CNPq (Grant #559912/2010-2).

References

- [1] M. Bauer, B. Bruegge, G. Klinker, A. MacWilliams, T. Reicher, S. Riss, C. Sandor and M. Wagner. “Design of a Component-based Augmented Reality Framework” In *ISAR’01, IEEE and ACM International Symposium on Augmented Reality*, New York, USA, 2001, pp. 45-54.
- [2] Y.Y. Di Wu and Y. Liu. “Collaborative Education UI in Augmented Reality” In *ETCS’09, First International Workshop on Education Technology and Computer Science*, Wuhan, China, 2009, pp. 670-673.
- [3] J.M.S. Dias, P. Santos, R. Bastos, L. Monteiro and R. Silvestre. “Developing and Authoring Mixed Reality with MX Toolkit”, In *ART 2003, 2nd IEEE*

- International Augmented Reality Toolkit Workshop*, Tokyo, Japan, 2003, pp. 18-26.
- [4] P. Grimm, M. Haller, V. Paelke, S. Reinhold, C. Reimann and J. Zauner. "Amire - Authoring Mixed Reality", In *The First IEEE International Workshop on Augmented Reality Toolkit*, Darmstadt, Germany, 2002, 2 pp.
- [5] A. Hampshire, H. Seitcher, R. Grasset and M. Billinghurst. "Augmented Reality Authoring: Generic Context from Programmer to Designer", In *OzCHI'06, 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments*, Sydney, Australia, 2006, pp. 409-412.
- [6] S. Irawati, S. Green, M. Billinghurst, A. Duenser and H. Ko. "An Evaluation of an Augmented Reality Multimodal Interface Using Speech and Paddle Gestures", In *Lecture Notes in Computer Science*, 4282, 2006, pp. 272-283.
- [7] H. Kato and M. Billinghurst. "Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System", In *IWAR'99, The 2nd Int. Workshop on Augmented Reality*, San Francisco, USA, 1999, pp. 85-94.
- [8] H. Kaufmann and A. Dunser. "Summary of Usability Evaluations of an Educational Augmented Reality Application", In *Lecture Notes in Computer Science*, 4563, 2007, pp. 660-669.
- [9] C. Kirner and R. Santin. "Interaction, Collaboration and Authoring in Augmented Reality Environments" In *SVR'09. XI Symposium on Virtual and Augmented Reality*, Porto Alegre, Brazil, 2009, pp. 210-220.
- [10] C. Kirner and T.G. Kirner. "Authoring Spatial Applications by non-Programmers with an Augmented Reality Tool" In *Web3DW 2010, IADIS Web Virtual Reality and Three-Dimensional Worlds Conference*, Freiburg, Germany, 2010, pp. 293-300.
- [11] C. Kirner. "ARAS-NP: Augmented Reality Authoring System for Non-programmers", Retrieved January 17, 2011, from <http://www.ckirner.com/sacra/>.
- [12] C. Kirner. "Q&A-AR: augmented reality racing game based on questions and answers", Retrieved February 27, 2011, from <http://www.ckirner.com/educa/>.
- [13] F. Ledermann and D. Schmalstieg. "APRIL: A High-level Framework for Creating Augmented Reality Presentations", In *VR'05, IEEE Virtual Reality 2005*, Bonn, Germany, 2005, pp. 187-194.
- [14] J. Looser, R. Grasset, H. Seichter and M. Billinghurst. "OSGART - A Pragmatic Approach to MR", In *ISMAR'06, 5th IEEE and ACM International Symp. on Mixed and Augmented Reality*, Santa Barbara, USA, 2006, 2 pp.
- [15] B. MacIntyre, M. Gandy, S. Dow and J.D. Bolter. "DART: A Toolkit for Rapid Design Exploration of Augmented Reality Experiences", In *UIST'04, 17th Annual ACM Symposium on User Interface Software and Technology*, Santa Fe, USA, 2004, pp. 197-206.
- [16] D. Schmalstieg, A. Fuhrmann, G. Hesina, Z. Szalavari, L.M. Encarnação, M. Gervautz and W. Purgathofer. "The Studierstube Augmented Reality Project", *Presence: Teleoperators and Virtual Environments*, 11(1), 2002, 33-54.
- [17] H. Seitcher, J. Looser and M. Billinghurst. "ComposAR: An Intuitive Tool for Authoring AR Applications", In *ISMAR'08, 7th IEEE and ACM International Symp. on Mixed and Augmented Reality*, Cambridge, UK, 2008, pp. 177-178.
- [18] S. Uchiyama, K. Takemoto, K. Satoh, H. Yamamoto and H. Tamura. "MR Platform: A Basic Body on which Mixed Reality Applications are Built", In *ISMAR'02, The First IEEE and ACM International Symposium on Mixed and Augmented Reality*, Darmstadt, Germany, 2002, pp. 246.

Author Biographies



Claudio Kirner Education: Undergraduate in Electrical Engineering - EESC/USP/Brazil - 1973; Master (M.Sc.) in Electronic Engineering (Hardware) - ITA/Brazil - 1978; Doctor (Ph.D.) in Systems Engineering and Computer Science - COPPE/UFRJ/Brazil - 1986; Visiting Researcher Professor in Virtual Reality - University of Colorado UCCS/USA - 1993-1995. Professional History: He is Professor at Federal University of Itajubá - UNIFEI, since 2009; He taught at Federal University of São Carlos - UFSCar, from 1974 until 1999, and at other universities, such as: University Center of Marília, Adventist University of São Paulo, Methodist University of Piracicaba and Federal University of Ouro Preto. He participated on the creation of the B.Sc. in Computer Science / UFSCar in 1974, and others; He advised 28 M.Sc. students and eight Ph.D. students; He coordinated several projects supported by Brazilian Agencies with grants of US\$750K, including two cooperation agreements with research institutes in Germany; He published over 220 scientific papers and 40 books and chapters as author and organizer. Interest Areas: Virtual Reality, Augmented Reality, Collaborative Environments, Human-Computer Interaction, Distance Education. Affiliation: Federal University of Itajubá - UNIFEI - Brazil. Email: ckirner@gmail.com



Tereza G. Kirner She has over twenty five years of academic experience, teaching and developing research on: software engineering and software quality; development and evaluation of virtual reality and augmented reality systems; and educational software applications. She has authored over a hundred refereed publications, advised students, and coordinated and participated of several funded projects. She is a Professor of undergraduate and graduate programs in Computer Science at the Federal University of Itajubá (UNIFEI), Minas Gerais, Brazil. Prior to his present position at UNIFEI, she worked in other important Brazilian universities (The Federal University of São Carlos - UFSCar, and the Methodist University of Piracicaba - UNIMEP, both in São Paulo state. Besides, she was a visiting researcher in two international software engineering centers, in the United States and Germany. She also served as Coordinator of Master Programs and B.S. Programs, as well as coordinated and participated of the creation of graduate and undergraduate courses. Interest Areas: Software Engineering, Collaborative Environments, Human-Computer Interaction, Virtual Reality, Augmented Reality, Distance Education. Affiliation: Federal University of Itajubá - UNIFEI - Brazil. Email: tgkirner@gmail.com