Development of an Immersive Challenge Racing Game for 3D-TV

Ana Grasielle Dionísio Corrêa, Alexandre Antonino Gonçalves Martinazzo, Erich Panzenboeck Lotto, Marcelo Archanjo José, Irene Karaguilla Ficheman, Roseli de Deus Lopes
Escola Politécnica da Universidade de São Paulo (EP-USP)
Laboratório de Sistemas Integráveis (LSI), São Paulo, Brasil
{anagrasi, am, irene, roseli} @lsi.usp.br; erichlotto@gmail.com; marcelo_archanjo@yahoo.com.br

***Abstract*—Television is most people favorite pastime. The way we watch TV has not changed since its invention 75 years ago. However, research in computer graphics and display technology favored the advances of TV platforms, such as interactivity and stereoscopy. Moreover device connectivity allows new ways of distributing content and applications, including films and games. Combining those innovations, 3D-TV is a new type of home entertainment that provides a rich and fun experience, innovating not only the way we watch movies and television series but also the way we interact. In this context, this paper presents the development of an immersive racing game for 3D-TV, discussing its design and some of the challenges and constraints related to available technology.**

Keywords*—*3D-TV; immersive games; entertainment;

I. INTRODUCTION

Research in digital communication and entertainment has been growing very fast in recent years [1], [2], [3]. On the other hand, provision of television services generates increased competition to attract the attention and interest of audience. As a result, television industry is constantly being challenged to innovate, whether by providing new content, or by interaction mechanisms in order to meet the changing expectations of their clients [4].

The evolution from Analog to Digital TV has brought many benefits to the public [3], [4]: high definition images, better sound quality and interactive applications that provide a richer and funnier experience. However, these are not the only advances. Research in computer graphics [5], [6], such as the creation of new algorithms for image processing, improvements of video compression techniques and development of high processing hardware capacity, along with advanced display technologies with stereoscopy triggered the development of new features in Digital TV: digital cinema and three-dimensional TV (3D-TV).

The success of three-dimensional stereoscopic movies has been clearly demonstrated. Available technologies have allowed substantial profit to producers and film companies [7]. For this reason, the Advanced Television Systems Committee (ATSC) in North America has created a planning team to examine the possible advantages and disadvantages, requirements and steps required to manufacture and deliver 3D-TV content [7]. Similar investigations are also being carried out by international standardization bodies such as the European Broadcasting Union (EBU), the International Telecommunication Union (ITU) and 3D Home Consortium [8].

Great interest in 3D-TV is the recognition that compared with conventional television (standard 2D), 3D technology improves significantly television entertainment value [9], [10]. Among the benefits are sense of depth [11], sense of presence [12] and naturalness [13].

Recognizing the opportunity for these benefits, the entertainment industry began investigating ways to make applications available to 3D TV [7], [14]. However, research in this field is still very recent. In this article, we present a brief summary of the main topics about the development of an immersive racing game for a 3D-TV set.

II. ADVANCES IN TV PLATFORMS

The use of graphics systems provides a natural way of interacting with computer systems. The ability to recognize 3D patterns allows humans to perceive and interpret image data quickly and efficiently [15]. Many research works in computer graphics have developed a number of techniques and devices that allow real-time three-dimensional interaction [16].

Recently, performances of embedded devices have increased [17]. This fact is a result of the evolution of hardware and software embedded in such devices. In particular, mobile devices are evolving such that they increasingly support complex applications. However rendering of three-dimensional environments in these devices is still considered a difficult task. Some of the characteristics that imply serious limitations to these types of systems are [17], [18]:

- Limited CPU processing power.
- Low memory storage capacity.

- Limited graphic acceleration performance.
- Low energy capacity.
- Absence of sophisticated developing and debugging software.

Several companies have began to develop microprocessor technology that enable 3D graphics rendering on mobile devices, such as NVIDIA ®, Qualcomm ®, Samsung ®, Intel ® and ARM ®. Although there has been a lot of progress in 3D applications support for mobile devices, this fact does not apply when it comes to digital TV. Research in this area is at an early stage so that there are few studies aimed at analyzing the potential of integrating these two technologies [17].

AMD ® made the first advances towards incorporating specific 3D graphics rendering in 3D-TV hardware and launched the AMD Xilleon ™ 220. It is a processor based on the MIPS architecture with support for 2D/3D graphics rendering and MPEG-2 video decoding. Later, Broadcom ® released the BCM7030 chip with a very similar architecture to the AMD processor. The BCM7030 supports OpenGL, Direct3D and BroadCastCL APIs. This chip comes integrated with some set-top-box families (devices embedding middleware). Broadcom was one of the first companies to release a chip that supported Digital TV embedded systems with efficient 3D rendering capabilities.

A. Application

Considering applications for 3D TV, some researchers have developed specific projects that addressed the challenges of using 3D technology. Some of these projects are related to areas such as Virtual Reality, Augmented Reality and 3D interfaces assisted by intelligent systems.

Reference [19] presents a platform with the ability to provide intelligent interaction within a Digital TV environment. The author discusses the use of a 3D virtual avatar to assist interaction between user and TV. Although the research focus is the provided intelligent service, the author discusses the advantages of using a virtual 3D avatar for user interaction [19]. The 3D avatar is based on a woman's face. System users could use the services of the avatar to access information about custom content.

In addition, reference [20] discusses the use of Augmented Reality in TV environments. As a case study, the author implements a game based on a sport called pellet. The game generates video content modified to include graphic scenes overlapping, offers multiple viewing angles and interactivity. The author uses the case study to evaluate the constraints, problems and potential use of integrating these two types of technologies. The author also discusses the difficulties in developing applications in relation to system design, hardware restriction and graphics APIs.

The work reported by reference [21] uses the standard video compression MPEG-4 for the description of three-dimensional scenes. The author uses the benefits of MPEG-4 and scene description language BIFS (Binary Format for Scene) to make a realistic 3D scene reconstruction. This reconstruction takes into account the movement of TV cameras and movement of characters in the environment. Reference [22] uses as a case study the 3D environments reconstruction related to sport events.

Another strategy for integrating 3D technologies in Digital TV environments depends on the extent of existing middleware standards to support the development and implementation of three-dimensional applications. Reference [2] analyzes this strategy in terms of a new level of interactivity within the TV environment. The author proposes a platform architecture for implementing these applications, based on European Digital TV standard. The author also analyzed the main available graphics APIs and proposed the inclusion of one of these APIs in the set of libraries that would be used by developers of interactive applications to generate 3D content. The author also discussed the platform resources where these native applications would run.

There are new challenges to produce games for digital TV. Initially one can think of two main problems [17], [23], [24]: hardware and usability. When thinking about the hardware level, we think of the set-top-box [17]. It is responsible for converting the received signals into a format that can be interpreted and presented by TV equipment. Currently, set-top-boxes have low memory and low processing power, leading us to work with limited resources, in other words, low memory capacity and physical space for storage and transmission of games. Therefore, it is quite difficult to develop more complex games, such as those that require more enhanced graphic detail such as 3D graphics. Regarding usability [24] one of the main problems is the remote control, because it is normally equipped with several buttons. The remote control does not have an anatomical nor ergonomic design for gameplay. Another problem is the infrared technology used in today's remote controls. It is functional only when pointed directly at the TV receiver. The strength and breadth of remote control signal depends on the amount and intensity of its LED transmitter.

Even with limitations regarding interactivity and interface, various studies on games for Digital TV can be found in literature pointing to methodological approaches and promising results [17], [23], [25], [26]. The reference [25] describes the development of games using Lua-NCL programming language. This language is consistent with the Ginga middleware, embedded in Brazilian set-top-boxes. The authors reported that the Ginga NCL emulator had some limitations and mainly played animations. Video is a feature that would make games more attractive and more graphically dynamic. Despite this obstacle, the paper presented a breakthrough in the field of Games for Digital TV, not specifically in the games nature, but in the details of used programming languages, images and sound formats, navigation and emulator.

The same problem regarding video was reported in another research [26]. The authors describe a car racing game with real-time data used to update the state of the game. According to the authors, car racing has always been a sport where many leading edge technologies have been tested and is a good candidate for combining audio and video content, real-time data and 3D game engine.

Reference [17] present a methodological process for creating games for the Brazilian System of Digital Television, since its pre-production to first demo completion. Reference [18] presents the development process of educational games for a commercial digital TV platform, while indicating their educational goal and application areas. The work of [23] proposes a framework for developing games for digital TV, which allows the developer to focus on content creation only, without concerns about technical issues or common tasks related to game development.

III. Game Development

This section describes the immersive 3D car racing game development. It is an action game where the user goal is to avoid obstacles on the road by steering a car. Players start with a score, but every time the car hits an obstacle the score is decreased. The gameplay ends when the score is equal or less than zero or if the player is able to complete the race with a positive score.

Action games require players to act quickly, and they also require accuracy and good timing to overcome challenges. It is perhaps the most basic of gaming genre, and certainly one of the most common. Action gameplay tends to emphasize on combat. There are many subgenres of action games, such as fighting games and first-person shooters. This game has a maximum length of 2'50" and is a single player game.

According to the documentation available at company's developer website, the 3D-TV platform is able to run Flash applications and Web applications. In this portal, the company also offers a Software Development Kit (SDK). When installed, the SDK emulator provides access to platform and programming interfaces for supported programming languages, including hardware's 3D resources and video playback. Flash Applications for 3D-TV can be developed with the help of the Adobe Flash CS4 (or higher) tool using the Action Script 2.0 language for animations and interactivity. Web applications can be developed with Eclipse and JavaScript, HTML and CSS. Figure 1 shows a layered architecture of the 3D- TV platform for programming in Action Script 2.0.

Developing applications with this platform presents some challenges for content adaptation. Although content can explore TV display quality, especially Full HD resolution, processing power and internal memory are limited. Thus, applications' features often need to balance quality display and processing capabilities. Still, this is a hardware limitation, and therefore will change with platform evolution.

The game was implemented with immersive 3D graphics capabilities, and the user needs glasses for passive stereoscopy. To control the game, the user must use the TV's remote control. It could be either the standard infrared control or the 3D-TV platform wireless remote control, which communicates with the TV set. The wireless remote control is able to send movement data and pressed keys using a proprietary protocol. Movement data is processed and transformed by the TV platform. Applications consider this data as a pointing device.

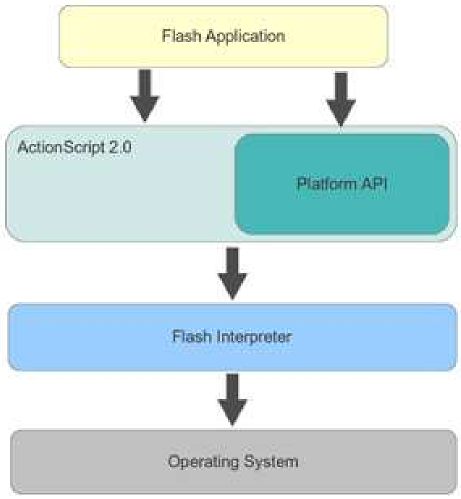

Fig. 1. SDK Layered Framework Organization

The game was not designed to compete with major console platforms, such as Xbox, Nintendo Wii or Playstation. The game is short and casual, just for entertainment purposes. We developed a game that was easy enough for users to be interested in continuing playing. Moreover, it becomes increasingly more challenging, so getting through the whole track is not as easy as the beginning.

Building an immersive 3D stereoscopic game without any Graphics Processing Unit (GPU) was difficult using current techniques. The TV set used in this study has an embedded GNU/Linux-based operating system, can be connected to the Internet and has a passive stereoscopic full HD display. It is also able to run interactive applications using Flash Lite 3.x or Web standards. Flash Applications can be developed using the Action Script 2.0 language to provide interactivity and animations, while Web applications can use JavaScript, HTML and CSS.

Here we list some application development constraints in this platform:

- The platform only provides one audio channel. That means background music cannot be played along with sound effects.
- Video resolution could be either 960x540 pixels or 1280x720 pixels. We opted for the latter in order to provide better image quality.
- Applications cannot be larger than 90MB.
- Due to TV set processing power, animation occupying large parts of the screen had to be avoided. Thus, every animation in the game had to be small, something close to 10% of the screen.

One of the platform advantages was to play videos directly on TV hardware. That's because the SDK provides ActionScript API to deal with TV resources (in this case, hardware video decoding features, including stereoscopic ones). These functions allow adjusting the exhibition mode for

3D videos, such as side-by-side and top-bottom, among others. This approach does not overload the CPU when playing videos, making that limited processor free to perform different tasks. Thus, it is possible to play a video animation on the entire screen, while small animations are implemented with Flash.

Considering all these constraints, we decided to develop a car racing action game with obstacles. It combines a 3D video (side-by-side) that shows a track with different obstacles (trestles, road cones, cars and trucks). Figure 2 shows road cones and vehicles on the track.

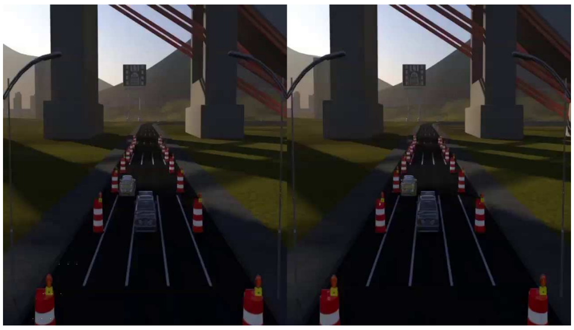

Fig. 2. Cones at track

The road has five tracks and a given obstacle will occupy one or more tracks while it is shown on the screen (Fig. 3). The 3D world was created with Autodesk Maya 2009. The textures were prepared with Adobe Photoshop CS5. For post-production and finishing we used Adobe After Effects CS5.

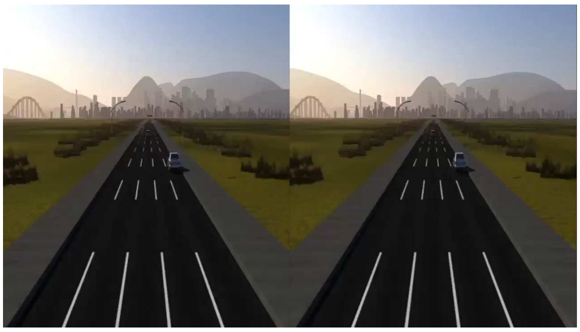

Fig. 3. 3D video images (left and right eye) side-by-side

The video playback is not dependent on user interaction, but it will heavily influence user decisions. If users reach the end of the road, then they win the game, otherwise the car breaks and the game ends.

Animation components (score and sprite of a car) are displayed over the video, and both are controlled by Flash. We adjusted the depth of the stereoscopic video to the positive space, making scene objects look farther compared to the screen plane. The animation layer is not stereoscopic and, as a result, the car will look closer to the user than any other objects on the road. This was necessary because the car is a 2D sprite applied to 3D video, so we could ensure visual consistency of 3D objects with 2D screen. Figure 4 shows how layers (3D video and Flash graphics) are composed in this game.

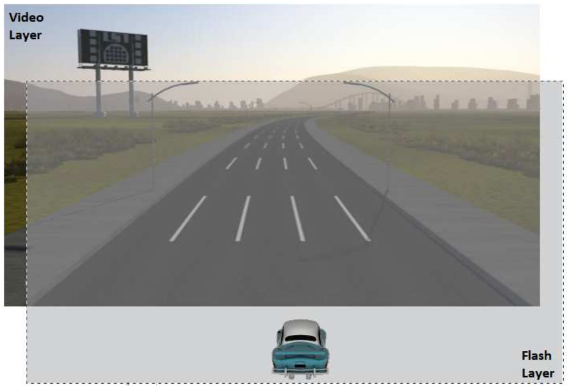

Fig. 4. Layer composition

Figure 5 shows a full screen image of the way the game is viewed on TV (except that on TV it is 3D) with the components in Flash.

Fig. 5. In the center, the sprite of the car; bottom left, game score indicator.

The car can occupy one of five possible tracks and sprites are changed according to car position, so the perspective remains consistent. Figure 6 shows five sprites side by side.

Fig. 6. Sprites side by side

During the game the sprite movement is controlled by remote control arrows (Fig. 7).

Fig. 7. Left and right arrows control car movements on the tracks

There was not enough processing power to build a game engine, and no API functions to create stereoscopic effects in Flash objects. In order to detect collisions between road objects and the car, we had to preprocess the video and generate a description of occupied tracks with timing.

An important contribution of this work is to promote the interaction of an object in Flash with 3D objects within the video. For this, we mapped all the obstacles in the video, relating their position in one of the five tracks to the time (in milliseconds) that they are displayed on screen. This is a concept equivalent to what occurs with subtitles in movies. We created an XML file containing this information. This can be seen in the following XML code fragment:

```
<time>
<start> 9333 </start>
<end> 9375 </end>
<points> -20 </points>
< track > 4 </track>
</time>
```

This XML fragment is directly related to an object on the screen, and is interpreted as follows: the object will appear on the screen at 9333 milliseconds of video playback time. The object is placed on the fourth track. A collision should subtract 20 points from the overall game score. That object will remain on screen until the playback reaches 9375 milliseconds. Every kind of obstacle is described with the XML structure. Obstacles are classified in the following way:

- Light: cones and trestles. 20 points are subtracted from total game score;
- Medium: cars. 30 points are subtracted from total game score;
- Heavy: trucks. 60 points are subtracted from total game score.

To generate that description we developed an image analysis algorithm with the OpenCV library (C/C++). The algorithm analyzes the video images frame by frame, searching for variations in gray tone on the track (any value greater than 40 in RGB format). To perform that search, nine checkpoints are established in each track. When an object crosses one of those points, the initial time frame is recorded. As the frames progress, the algorithm will analyze these points to identify frames where the track reverts to the standard shade of gray. Thus, we can identify when an object is at a given time and its track, and then generate XML file.

We also generate a bitmap identifying the points analyzed. For example, Figure 8 shows a frame at a given time, where each track contains a set of reference points to detect the objects. The obstacle is in the 4th track, and these reference points (located in the bottom of the screen) detect it when crossing the area between those points.

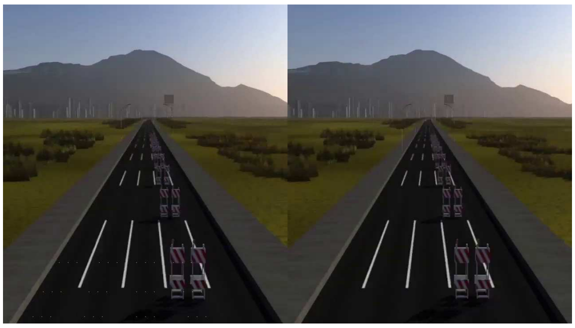

Fig. 8. Frame analyzed by image analysis algorithm that identified obstacles on track 4

IV. PRELIMINARY TESTS

The immersive racing game was shown to the public (mostly college students) during the First Fair Innovation & Entrepreneurship at University of Sao Paulo. It was an interesting time to check the acceptability of the game by the target audience. There were no scientific assessments, but rather, observation and perception. The test was conducted in a space containing a TV set with the running game (Fig. 9).

The game became the most attractive booth. Users watched TV with the game running and approached to play. The players stood together and took turns to use the remote control and 3D glasses. Each user played until reaching zero score and then passed the remote control to the next user on line. Two people have assisted the team by instructing users how to play and controlling the waiting line. Each game round lasted an average of one minute, sometimes a little more depending on the player. We used 8 passive glasses so that several people could follow the game while one player used the remote control. This caused a crowd around the booth and interests of other people to try the game.

Fig. 9. Game test setup.

Most users struggled to reach the end of the game. A small percentage managed to finish the game with a single attempt. The majority needed three or more attempts to successfully reach the end of the road. A very small amount of users quit the game before reaching the end (success or failure). We observed that cancellation occurred due to pressure of the public waiting for their turn.

V. FINAL REMARKS

This paper presented the development of the Immersive 3D Challenge Race game. Working with a Digital TV was enriching for the team, stressing that research in this area must continue. We also point out that layering a 3D video and Flash animations is an innovative technique. We have developed a low demand processing power approach that enabled us to create a game on a constrained platform.

In the case of this project, the main difficulties were: trying to use passive stereoscopy on TV without GPU, the small animation capability, limited video resolutions and single audio channel. Despite of these difficulties, the game has been implemented, and preliminary tests with target audience indicate the attractiveness and interest of the public, as well as easy to use remote control.

We plan on conducting additional usability tests with users of different age groups and genders. Other tests that will be conducted refer to the use of TV in a group, because this equipment is inherently a collective unit for use in homes.

Unlike a computer, which is designed for individual use, the TV was designed to be used in small groups. Often use is with family members. And with the possibility of having a connected TV, these games can be expanded exploring the return channel. It would be possible, for example, to enable collaboration of players in different locations with different TVs or even other devices available on the same network.

REFERENCES

[1] E. Tsekleves, J. Cosmas, A. Aggoun, J. Loo J. "Converged Digital TV Services: The Role of Miccleware and Future Directions of Interactive Television". International Journal of Digital Multimedia Broadcasting, 2009, pp. 1-19.

[2] P. Cesar, K. Chorianopoulos, "The Evolution of TV Systems, Content, and Users Toward Interactivity". Journal Foundation and Trends in Human-Computer Interaction, vol. 2, n.4, 2009, pp. 373-395.

[3] G. Doyle. "From Television to Multi-Platform. Less from More or More for Less?" International Journal of Research into New Media Technologies, vol. 16, n. 4, 2010, pp. 431–449.

[4] P.C. Murschetz, T. Evens, "Smart TV in Germany – How Does Convergence Impact on Market Structure and Business Model Venturing in Digital Television Broadcasting?" Available from: https://biblio.ugent.be/input/download?func=downloadFile&recordOId=3224401&fileOId=3224402 [Accessed 03 June 2013].

[5] A. Redert, M.O. Beeck, C. Fehn, W. Ijsselsteijn, M. Pollefeys, L. Van Gool, E. Ofek, I. Sexton, P. Surman. "Advanced three-dimensional television system technologies". First International Symposium on 3D Data Processing Visualization and Transmission, 2002, pp. 313-319.

[6] C.E.I. Erdem, F. Ernst, A. Redert, E. Hendriks. "Temporal stabilization of Video Object Segmentation for 3D-TV applications. Signal Processing". Image Comunication, vol. 20, n. 2, 2005, pp. 151-167.

[7] W.J. Tam, F. Speranza, S. Yano, K. Shimono, H. Ono. "Stereoscopic 3D-TV: Visual Comfort". IEEE Transactions on Broadcasting, vol. 57, n.2, 2011, pp. 335-346

[8] P. Anguera, D. Vega, J. Morgade, M.M. Vélez. Transmission of 3D Video over Broadcasting. 3D-TV System with Depht-Image-Based Rendering: Springer New York, 2013, pp. 299-344.

[9] S. Yano. "Experimental stereoscopic high-definition television". Displays, vol. 12, n. 2, 1991, pp. 58-64.

[10] P.J.H. Seuntëns. "Visual Experience of 3D TV". Thesis presents to University of Technology and Philips Research Eindhoven: Netherlands, 2006, 135p.

[11] A. Berthold. "The influence of blur on the perceived quality and sensation of depth of 2D and stereo images". ATR Human Information Processing Research Laboratories Technical Report, TR-H-232, Kyoto, Japan, 1997.

[12] W.A.I. Jsselsteijn, H. Ridder, R. Hamberg, D. Bouwhuis, J. Freeman. "Perceived depth and the feeling of presence in 3DTV". Displays, vol. 18, n. 2, 1998, pp. 207-214.

[13] R.G. Kaptein, A. Kuijsters, M.T.M. Lambooij, W.A. IJsselsteijn, I. Heynderickx. "Performance evaluation of 3D-TV systems. Proc. Electron". Imaging Image Quality Syst. Perform, vol. 6808, pp. 1-11.

[14] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, C. Zhang. "Multiview Imaging 3D-TV". IEEE Signal Processing Magazine, 2007, pp. 10-21.

[15] J. Foley, D. Dan, S.K. Feiner, J.F. Huger. Computer Graphics: Principles and Practice. Boston: Addison Wesley, 1997.

[16] T. Möller, A. Haines, Real-Time Rendering. Massachusetts: A.K. Peters, 2002.

[17] A.N. Silva Jr., A.C.S. Souza, L.C.M. Santos, R.L. Sampaio, P.O. Raimundo. "Desenvolvimento de Jogos para o Sistema Brasileiro de TV Digital". VIII Simpósio Brasileiro de Jogos e Entretenimento Digital, 2009.

[18] A.G.D. Corrêa, A.A.G. Martinazzo, L.C. Biazon, M. Archanjo, V. Venancio, I.K. Ficheman, R.D. Lopes, R.D. "Jogos educacionais para TV digital interativa". Revista Trilha Digital, vol. 1, n. 1, 2013, pp. 38-50.

[19] S. Maad. "MARILYN: a novel platform for intelligent interactive TV (IITV)". In: J.A. Jacko, C. Stephanidis (Org.), Human-Computer Interaction, Theory and Practice (Part 1), Lawrence Erlbaum Associates, Mahwah, NJ, vol. 1, 2003, pp. 1041-5

[20] T.D. Kammann. Interactive Augmented Reality in Digital Broadcasting Environments. Document of Diplom-Informatiker im Studiengang Computervisualistik, Universitat Kolenz Landau, Koblenz, 2005.

[21] E. Boyle. "The creation of MPEG-4 content and its delivery over DVB infrastructure". Proceedings of the first Joint IEI/IEEE Symposium on Telecommunications Systems Research, Dublin, Ireland, 2005.

[22] C. Malerczyk, K. Klein, T. Weibeisek. "3D reconstruction of sports events for digital TV". Proceedings of the 11th International Conference in Central Europe on Computer Graphics, Visualization, and Computer Vision, Plzen, Czech Republic, 2003.

[23] D.C. Barboza. "Ginga Game: a framework for game development for the interactive digital television". VIII Symposium on Brazilian Games and Digital Entertainment (SBGAMES), 2009, pp. 162-167.

[24] R.G.B. Luis, R.M.C. Segundo, T.A. Tavares. "Jogos eletrônicos na TV Digital brasileira: gêneros e desafios para esta nova plataforma". IX Symposium on Brazilian Games and Digital Entertainment (SBGAMES), 2010.

[25] M. Minho, A. Santos, L. Bridi, L. Alves, M. Moret. "Jogos Eletrônicos para TV Digital: relato de experiência". VIII Brazilian Symposium on Games and Digital Entertainment (SBGAMES), Rio de Janeiro, 2009.

[26] S. Gibbs, M. Hoch, L. Rafey, S. Wang. "Broadcast Games and Digital Television". ACM International Game Technology Conference, Hong Hong, 17-21 January. Available from: http://www.khm.de/~micha/Publications/dtvGame.pdf. [Accessed 11 June 2013].