

# Augmenting Augmented Reality with Pairwise Interactions: The Case of *Count Luigi Ferdinando Marsili* Shooting Game

M. Roccetti, G. Marfia, A. Amoroso, S. Caraceni  
Dept. of Computer Science  
University of Bologna, Italy  
Mura A. Zamboni 7, 40127 Bologna  
{roccetti, marfia, amoroso, caraceni}@cs.unibo.it

Angelo Varni  
Dept. of History, Geography and Anthropology  
University of Bologna, Italy  
P.za San Giovanni in Monte 2, 40124 Bologna  
angelo.varni@unibo.it

**Abstract**—Thanks to an ever increasing processing power that can today support advanced graphical capabilities within high bandwidth mobile devices, augmented reality technologies are now ubiquitous, finding profitable uses in a wide variety of scenarios. Augmented reality principles, in fact, can be beneficially put to good use within many types of mobile applications: for example, augmented city guides can properly display additional virtual objects that show how a monument was at the time of its construction (e.g., the Colosseum with the inclusion of its missing parts), or augmented games can add fictitious characters and objects to real world settings in proper places (e.g., Lara Croft in search of Bartoli in Venice). Nonetheless, the totality of such augmented mobile applications, to the best of our knowledge, limit the interactions that may occur to those that can happen between a single mobile user and the objects that are displayed on his/her mobile platform screen. In this work, we show that it is possible to devise a new interaction paradigm, augmenting augmented reality apps with the interactions that may occur between multiple users via the virtual objects displayed on multiple mobile platforms. In particular, we applied such ideas to the devise of a shooting game set in the 17<sup>th</sup> century to celebrate the 300<sup>th</sup> anniversary since the foundation of the Academy of Science of Bologna by Count Luigi Ferdinando Marsili. The paper describes and discusses the design choices and the technical issues at the basis of our system.

**Keywords** - *Augmented Reality; Interactive AR ; Mobile AR; Luigi Ferdinando Marsili; Academy of Science of Bologna.*

## I. INTRODUCTION

A need for reality (as opposite to fiction and virtuality) is again pervading our society in many different ways, ranging from a plethora of big brother reality shows that have invaded broadcasting media since the beginning of the third millennium, to new forms of physical and tangible interaction with computer games that have determined the successful introduction of gaming consoles as the Nintendo Wii. In this fast flowing world where technology and media are taking on new forms to better interact with people, the time is probably ripe for a new form of reality, i.e. Augmented Reality (AR), that aims at exceeding the information conveyed by reality itself [1]-[8]. AR, in fact, may be seen as a kind of *super* reality where an application designer, solely constrained by the space

where a user moves, can freely place virtual objects that either were there some time in the past, or will be there some time in the future, or that exist only in his/her imagination.

The binding of reality to *virtuality* of AR-based applications has, since the beginning of the existence of such type of technology, put it at the center of the attention of many different designers that regarded it as a basis for several different and useful applications. In marketing, for example, AR principles have been successfully applied to show customers how certain products could potentially appear or be used. In sports telecasting, it is now common to add virtual objects to the real playing field (e.g., virtual yellow first-down line superimposed on an actual football field), in order to explain what is happening to who is watching a game. Even in the devise of military technology for warfare, AR is now being considered to provide more intuitive and faster interfaces to aircraft pilots, for example utilizing helmet mounted displays, which allow pilots to look through aircraft walls as if they were flying and floating in the sky, like superman.

But the great news of these recent years is the blending of AR and mobile technologies. As users move, for example, they can watch through their smartphone or tablet displays and observe an augmented world that meets their requests for information or entertainment. All this can be done thanks to the integration on mobile devices of geographic referencing technologies such as GPSs and gyroscopes that provide AR applications with updated and precise estimates of the position and direction where the screen is pointing. However, also in this new and dynamic scenario of use, application designers have thought and engineered interactions that occur solely between a single user and an augmented world displayed on his/her respective mobile device. Now, the time is ripe for a new step forward: users could interact not limiting themselves to their *private* AR world, but together, as if they were part of a community, interact with each other and with the virtual objects that populate their respective common *virtual spaces*. Clearly, such type of paradigm requires the implementation of advanced means of communication between AR worlds that are displayed on different mobile devices as well as the definition of how multiple users can interact with the same virtual objects that are displayed, from different perspectives,

in front of them, in a synchronous and consistent manner. This can be accomplished for example by *augmenting* augmented reality technologies with the network that is in place and already active between connected mobile devices, to allow interactions among multiple users via a given virtual world with its objects.

To provide a practical and factual demonstration of how all this can take place, we designed a new game that implements such principle: while involved in a game, two given players share a single virtual world, which they observe from different perspectives, and through which they interact, seeing the effects of the actions performed by both players. The opportunity for devising, implementing and experimenting this particular game came from the desire of celebrating the 300th anniversary since the foundation of the Academy of Sciences of Bologna by Count Luigi Ferdinando Marsili. Marsili, who founded the institution in 1711, was very active in the late 17th and early 18th centuries as a soldier (well known for the design of innovative fortifications, cannons and howitzers) as well as a scientist. With this in view, we designed a shooting game, where two players, armed with a virtual howitzer, can aim at each other through their smartphone screen and finally shoot. The game is organized in rounds, where during each round only one player can shoot, while the other can only move within a limited area to try to avoid being hit by the fired shell. Once one of the two players shoots by tapping on his/her smartphone screen, the fired shell leaves the smartphone and is visualized on a set of screens that are placed between the two players. If the shooter aimed right, the opponent is shot; in any case the information of a miss or a hit is finally displayed synchronously on the smartphones of both players. Clearly, all this requires: (a) a way of determining whether the position where the shooter tapped on his/her smartphone matched the position where the opponent was at the time of shooting, (b) the communication to both devices of the result (miss or hit), (c) a common view of each game phase from both smartphones, and, (d) the synchronized visualization of a shell departing from a smartphone, entering the first screen, moving from screen to screen to finally reach the opponent's side. All this is discussed in our paper where design choices and technical issues at the basis of our system are discussed. The remainder of the paper is organized as follows. In Section 2 we describe the historical background and the chosen visualization models chosen. We then provide an overview of the system in Section 3. All technical issues and employed solutions are discussed in Section 4. We finally conclude with Section 5.

## II. HISTORICAL BACKGROUND WITH VISUAL REPRESENTATION

The aim of the augmented reality system described in this paper is to become part of a temporary exhibition that will be held to celebrate the 3rd centenary since the foundation of the Academy of Science of Bologna. This temporary exhibition will take place in the so-called *9th Centenary Museum* of the University of Bologna that was built with the aim of providing a permanent exhibit displaying the intangible heritage of the nine centuries of history of that University. This Museum is comprised of three different sections: the *Time Chambers* which host eight exhibits where it was possible to explore the main issues related to the first eight centuries of history of the

University of Bologna, the *Ritual Chamber* where a specific document, termed the *Magna Charta Universitatum*, is exposed, and finally the *Space Chambers* (also termed the Aleph rooms), where our temporary exhibition is going to take place. Specifically, the two Aleph rooms are two circular rooms, connected by a straight aisle. Each one is equipped with nine video screens that can rotate on a carousel (as shown in Figure 1).



Figure 1. The two Aleph rooms (by courtesy of University of Bologna).

Within the context provided by these two rooms we designed an interactive AR-based shooting game, with the specific aim of illustrating both a part of the life of the Count Luigi Ferdinando Marsili and also the specific episode that gave birth to the Academy of Science of the University of Bologna that was founded on December 12<sup>th</sup> 1711. Within his long and adventurous life, in fact, there is a particular episode at the basis of the rise of the Academia. This traces back to the siege of the city of Breisach (1703) that capitulated after a long period of resistance. A wooden model of the Breisach siege is shown in Figure 2, where the fortified town with cannons, howitzers and mortars is represented. Unfortunately (or fortunately), Marsili (who was an officer in charge of defending the city) was considered as responsible of this defeat and was hence stripped of all honors and dismissed as an officer. After this episode, Marsili returned back to his city (Bologna) and devoted all the remainder of his life to science, mainly as a naturalist, thus commencing the scientific activities of the aforementioned Academy [9].

As to the visual information we decided to display during our shooting game, it is important to notice that our aim was that of conveying a precise message: war, a topic which can be undoubtedly interpreted as negative, when abandoned, may free new energies that, in turn, can give rise to the blossom of positive expressions for human beings, like natural science, for example. Owing to these motivations, the information visualized during our shooting game alternate between those drawn from the military activities performed by Marsili and

those concerning his professional life as a naturalist. In particular, these represent the corals, fishes and fossils that were collected by Count Marsili during the scientific studies that followed his military career. In particular, as the visual representation of our game takes place both on the display of the device of each player and also on the screens of the Aleph rooms in the Museum, we distributed and animated them as explained in the following.



Figure 2. Model of the city of Vieux-Breisach, XVIII century, polychrome wood (by courtesy of University of Bologna).

- *Player's Device Display.* Two alternative interfaces were chosen to be visualized on the display of the game device (a Smartphone) used by both players: a minimal one and an heavier, ancient-looking one, respectively. With the minimal one, the shooter has no gun shown. The interface shows instead a round animated circle that represents a *life counter*, indicating to the player whether he/she is still alive or has been shot. In the ancient-looking one, the shooter has a bombard at the bottom of the display. The opponent instead has at the bottom of its screen a part of the Breisach fortification, and an hourglass as a life counter.
- *Screens of the Aleph Rooms.* On the initialization phase, all those screens display images chosen from the Marsili's naturalist collection, integrated by captions inviting players to commence the game (at the same time, an ambient sound is played as a musical background). After the initialization phase, all those images fade down replaced by the visualization of military architectures that are displayed as a visual background. Upon shooting, a cannonball is visualized running through the screens of the Aleph room where the shooter stands, lasting for three consecutive loops. Upon completion of the third loop within the Aleph room where the shooter is, the cannonball disappears from those screens and immediately re-appears on the screens of the other Aleph room where the target is supposed to be. The ball runs again for three other loops across the screens of that room. On completion of this final loop, new images inspired to the Marsili's activity as a naturalist

appear on all the eighteen screens of the two Aleph rooms to conclude the game visualization.

Thus, in some sense, the cannonball running through the eighteen screens of the Aleph rooms, with a background given at first by some military scenes and then by some naturalistic images, should resemble the adventurous trajectory of the life of the Count. Obviously, besides celebrating the third centenary of the birth of the Academy of Science with a new, and hopefully exciting, gaming experience, we are confident that developing an AR-based game, that unites playful shooting activities with the visualization of more serious aspects related to military architectures and to natural sciences, may be helpful to persuade people (specifically youngsters) to visit the entire Museum, especially those sections which are devoted to its founder [10].

### III. OVERVIEW OF THE SYSTEM

We first briefly sketch how the game evolves based on a finite state model and then discuss the software architecture of the system.

As to the game, the basic idea is to allow two players (shooter and opponent) to interact each other via an augmented reality space, using as their mobile devices two Smartphones. Each player stands at the center of a given Aleph room, with the screen showing images related both to the game going on and to the life of Marsili, as previously discussed. Each player, hence, grips an Smartphone, and commences playing the game using that device. Ideally, the game represents two armies that bomb each other by means of cannonballs. This resembles those warfare situations taking place during the Marsili's age. To better understand how the game evolves we now provide its finite state model.

#### A. Finite State Model

Figure 3 summarizes the finite state model of the game. The first state is the registration of the players. Players register by framing a printed tag with the Smartphone camera. The game evolves through consecutive rounds. At each round, each player can play as either the *shooter (attacker)* or the *opponent (defendant)*. At the beginning (first round), the role of each player is decided based on the chronological order of his/her registration: the first to complete the registration is the shooter, while the second performs as the defendant. With each new round, players swap their roles. During a round, both the shooter and the opponent watch each other through their Smartphone. While the shooter can aim at the defender using the Smartphone, the opponent has on its screen information on his/her state (dead or alive). Obviously, the opponent can move, within in the limited space of the Aleph room, to avoid the shot. Eventually, if the shot hits the opponent, the attacker scores a point. As mentioned, at the end of each round, the shooter and the opponent swap their roles, and the game goes for another round. After a given (typically three) number of rounds, the game quits, and the player that got the higher score wins.

### B. System Architecture

The game is implemented based on the functional architecture shown in Figure 4. All takes place within the two Aleph rooms.

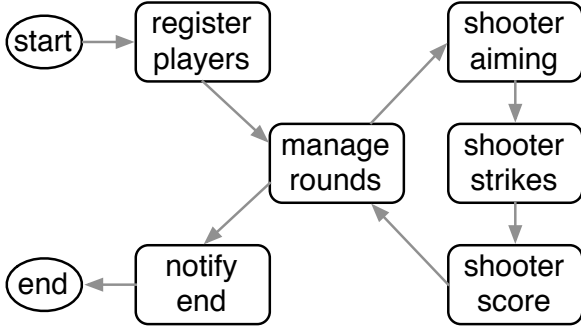


Figure 3. Finite state model of the game.

Each room hosts: a carousel mounting nine screens, a server and an Smartphone. The server in each room controls the screens of that room, and also runs a few software processes that implement part of the game. While the two servers are connected to each other, only one of them (termed *Server 1*) is connected to the Internet through a wired connection. The two Smartphones are connected to the Internet by means of a wireless connection (represented in the Figure by dotted arrows). The Internet allows both the *Server 1* and the Smartphones to access the basic service provided by the *Junaio* AR platform.

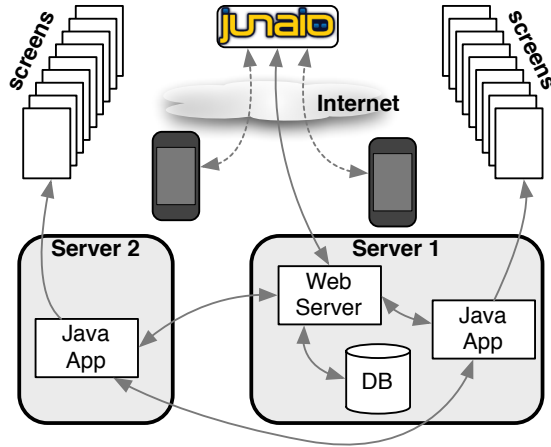


Figure 4. Functional architecture of the system.

Server 1 runs a web service that manages the game logic, while storing in a database the global game state. Server 1 also runs a Java application that displays on the nine screens the images that visualize the evolution of the game phases, orchestrated and synchronized as discussed in Section II. Server 2 runs an identical Java application that orchestrates the information visualization on the screens of the room where it is placed. When the shooter aims at the opponent, its Smartphone screen displays a trigger through which the opponent can be individuated. The defendant instead has on his/her screen an icon showing its state. It is the responsibility of the web service running on *Server 1* to properly manage the visual information

on each different Smartphone display, according to the role of the different player. As soon the shooter shoots, all the screens in its room display a cannonball that moves from screen to screen and overflies several historical locations inspired to the military career of Marsili. After three loops in the attacker's room, the cannonball *flies* into the defendant's room. As in the previous case, the screens show the cannonball overflying Marsili's historical images. At the end of the round, both the Smartphones and the screens in the two rooms display information regarding the result of that game round. Most important to notice, before concluding this Section, is the fact that it is *Junaio* that understands if the opponent has been shot or not. In fact it is *Junaio* that notifies our web server with the result of the shooting event. Reacting to these notifications, our server updates the game state, decides the information to be displayed on the two Smartphones (via *Junaio*), and finally orchestrates the images to be shown on the screens of the two Aleph rooms (via the Java applications). Obviously, all this is made possible by the fact that both Smartphones run a client that delivers the AR service provided by *Junaio*. What still needs an explanation is how the AR service provided by *Junaio* may allow players to shoot each other. This is explained in the following Section.

### IV. TECHNICAL ISSUES WITH SOLUTIONS

As anticipated before, our AR-based game is based on the use of: (a) the *Junaio* platform (b) two Smartphones and (c) a Java application. Needless to say, *Junaio* plays the role of the AR front-end with which players directly interact. Nonetheless, each player also interacts with his/her opponent (via *Junaio*), even if this is NOT *natively* supported by that platform. Basically, two were the problems that we sorted out to allow players to interact each other via *Junaio*: (i) associating a tap on the Smartphone screen with a real object (the defendant) subjected to the shooting and (ii) allowing this object (the defendant) to move in the reality, while permitting to the shooter to exactly identify his/her position in the real space. All this was performed through the use of two schemes, the former being based on an original use of the so called "points of interest" provided by *Junaio* (i.e., the points that are geo-referenced and associated to a given virtual object) and a latter based on the use of the Smartphone accelerometer. In the following, after an explanation on how *Junaio* works, we will provide details of the mechanisms we devised to allow pairwise interactions between players.

#### A. The *Junaio* Platform

*Junaio* is an AR platform specifically developed for mobile devices. It can run on all the major mobile operating systems (including Android and iOS) and, contrasted against possible alternatives, provides the additional functionality of utilizing Latitude/Longitude/Altitude (LLA) markers. Moreover, *Junaio* has a certain degree of flexibility that makes it adaptable to a developer's needs, especially compared to other competing AR platforms targeting the same markets. Basically, the service provided by *Junaio* plays the role of a proxy with the external world, as it acts as the point across which information flows from an application web server to the AR browser, and vice versa. All the virtual objects developed for a given mobile *app*, in fact, are stored on an external web server that is then

contacted by the *Junaio* service center, according to the requests made by the AR browser, following a client/server communication paradigm. Hence, *Junaio* retrieves so-called Points Of Interests (POIs) and virtual objects from an application server and then sends them to the AR client every time the mobile device triggers an event that requires a refresh of their display. As we shall see in the following, this scheme is at the basis of those mechanisms that allowed us to associate a tap on the phone to an area shot in the opponent's position.

### B. Pairwise Shooting Interactions

We devised an original mechanism to associate a tap on the shooter's device screen with the position where the opponent is considered to be during a shooting event. This was done based on the use of two virtual objects displayed at pre-defined POIs. The *Junaio* platform permits to associate events to taps on virtual objects that are displayed on the mobile device screen. When tapping on a virtual object, this information is sent to the *Junaio* service that, according to the virtual object that has been tapped, triggers an event. We have hence created two POIs, each of them corresponding to a different virtual object. Of the two virtual objects, one occupies the leftmost half of the real space of the Aleph room of the defendant, as seen on the shooter's device screen, while the other occupies the rightmost half of that space. This way, we established a relationship between the virtual object displayed on the shooter's device screen and the real space where the opponent is. In essence, if the shooter taps on the leftmost half (object) of his/her device screen while the opponent is in the leftmost half of the Aleph room, he/she results to be shot. Otherwise the opponent is missed. The same scheme holds on the rightmost side. As a result, when shooting, the shooter taps on one of the two halves of his/her device screen thus triggering a message sent to the *Junaio* service center. The shot information within this message is compared with the defendant's position provided by the web server to *Junaio*, in order to sort out whether the defendant has been shot or not. Clearly, this works well only if each mobile device communicates in real-time its position to *Junaio*. We will see how this is performed in the next Subsection.

### C. Identifying the Opponent's Position with an Accelerometer

The Smartphone accelerometer plays a central role in tracking where the defendant is while the shooter is shooting. The problem we had to address in this case, in fact, was that of matching the position at which the attacker was shooting with the position where the defendant was at that precise time. Many issues rose from the very limited precision of the accelerometers that Smartphones (like the iPhone, for example) carry, making this part of the system one of the most challenging ones to deliver. Trying to exploit physics equations (i.e., finding position as an integral of velocity, which in turn is found as an integral of acceleration) proved to be useless, given the high noise with which acceleration samples were fed by the sensor. For this reason, we opted for an alternative heuristic that, within our context of use, resulted to be efficient. This is as follows. Aleph rooms are circular rooms of a diameter of three meters connected by a straight aisle. Hence, a player, when moving perpendicularly to the corridor, can do at most one or two steps in one direction or in the other (right or left).

Thus, the objective of our positioning system was not that of having a precise estimation of where the player is, but a precise estimate of the direction where he/she is rapidly moving (i.e., right or left). We hence developed a software mechanism able to identify if a player is moving right or left, based on the consideration that the most reliable information that the accelerometer delivers is given by the value of acceleration (above a given threshold) when moving in a certain direction. Summarizing, if the attacker shoots aiming at the leftmost (rightmost) half of his/her device screen while the defendant moves rapidly towards the rightmost (leftmost) side of the room, the defendant can avoid being shot.

## V. CONCLUSIONS

We discussed on how to augment the interaction degree of AR mobile apps by exploiting interactions occurring between multiple users and virtual objects [11]. We applied such ideas to the design of a game that celebrates the foundation of the Academy of Science of the University of Bologna.

## ACKNOWLEDGMENTS

Our acknowledgements go to the DAMASCO (FIRB) and ALTER-NET (PRIN) projects. We are indebted to the students of the CS Class Multimedia Syst. & Apps who assisted us during the development of our system and to SERDATA for supplying much of the exploited hardware equipment.

## REFERENCES

- [1] T. Lee, T. Hollerer, "Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking," in *Proc. of the 11th IEEE International Symposium on Wearable Computers*, pp. 1-8, 2007.
- [2] T. Höllerer and S. Feiner, "Mobile Augmented Reality," in Karimi, H. and Hammad, A., (Eds), *Telegeoinformatics: Location-Based Computing and Services*. Taylor and Francis, 2004.
- [3] D.W.F. van Krevelen, and R. Poelman, "A Survey of Augmented Reality Technologies, Applications and Limitations," *The International Journal of Virtual Reality*, 9(2), pp.1-20, 2010.
- [4] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, B. MacIntyre, "Recent Advances in Augmented Reality," *IEEE Computer Graphics and Applications*, 21(6), pp. 34-47, 2011.
- [5] B. Bell, S. Feiner, and T. Höllerer, "View Management for Virtual and Augmented Reality," in *Proc. of the 14th Annual ACM Symposium on User Interface Software and Technology*, pp. 101-110, 2001.
- [6] A. Henrysson and M. Ollila, "UMAR: Ubiquitous Mobile Augmented Reality," in *Proc. of the 3rd ACM International Conference on Mobile and Ubiquitous Multimedia*, pp. 41-45, 2004.
- [7] A. Hamidian, C.E. Palazzi, T.Y. Chong, M. Gerla and U. Körner, "Exploring Wireless Mech Networks for Collaborative Augmented Reality Environments," *The International Journal of Virtual Reality*, 9(2), pp. 21-30, 2010.
- [8] J. Rekimoto and Y. Ayatsuka, "CyberCode: Designing Augmented Reality Environments with Visual Tags," in *Proc. of the ACM Conference on Designing Augmented Reality Environments*, pp. 1-10, 2000.
- [9] W. Tega, (Ed.), "Guida al Museo di Palazzo Poggi," *Scienza ed Arte*, Editrice Compositori, Bologna, 2005, p. 132.
- [10] S. Caraceni, "A Polymorphic Museum," in *New Realities: Being Syncrctic*, Ascott/Bast/Fiel/Jahrmann/Schnell (Eds.), Springer, 2008, p. 63.
- [11] M. Rocchetti, A. Semeraro, G. Marfia, "On the Design and Player Satisfaction Evaluation of an Immersive Gestural Game: the Case of Tortellino X-Perience at the Shanghai World Expo," in *Proc. of 29th ACM Conference on Design of Communication*, 2011, pp. 1-6.