AUGMENTED REALITY IN A SHOOTING GAME WITH REAL ROBOTS

Submitted By

Mujeeb ur Rehman	(14-CP-35)
Jawad Qammar	(14-CP-87)
Waleed Afzal	(14-CP-97)

Project Supervisor

Dr. Waqar Ahmad

DEPARTMENT OF COMPUTER ENGINEERING
UNIVERSITY OF ENGINEERING AND TECHNOLOGY TAXILA

July 2018

ABSTRACT

Augmented reality (AR) technology, usually incorporated into special headsets, eyewear, or projections, superimposes data or graphics over real-world images and uses sensors and cameras to capture the operator's motions for feedback and control. Until now, AR's primary application has been in gaming. But as the technology has become commoditized, it's now finding a surprising new role in robotics research, and it may soon have a significant impact on manufacturing and logistics automation, and eventually even home and service robots. Focusing on primary application of AR's, we are developing a game that you can play with your mobile device or a computer, being on the control of real robots. AR changes the game, it allows a human operator to get inside the robot's head, so to visualize. The operator uses AR to control the robot using graphical interface, giving the robot precise instructions simply by doing the tasks he or she wants the robot to emulate. In a nutshell this game includes, video streaming from a robot mounted camera to your controlling device with some overlay layers included for rendering game features (like explosions). Laser gun and shields (they are infrared counterparts to be completely safe). Motors and wheels controlled by your device. RFID tags used for power-ups on battle field (a RFID reader is included on each robot). Use Wi-Fi for communicating robots and players. One robot may act as an Access Point to provide outdoor fun. This game is supposed to be played with 2 or more robots, one player controlling each one.

UNDERTAKING

I certify that project work titled "*Augmented Reality in A Shooting Game with Real Robots*" is my own work. The work was not, in whole or in part, presented elsewhere for evaluation. When the material was used by other sources, it has been properly recognized / reported.

Signature of Student

Mujeeb ur Rehman (14-CP-35)

Jawad Qammar (14-CP-87)

Waleed Afzal (14-CP-97)

ACKNOWLEDGEMENTS

"I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them. I thank my God for providing me everything that I required in completing this project. I am highly indebted to Dr. Waqar Ahmad for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project. I would like to express my gratitude towards my parents & for their kind co-operation and encouragement which help me in completion of this project. I would like to express my special gratitude and thanks to industry persons for giving me such attention and time. My thanks and appreciations also go to my classmates in developing the project and people who have willingly helped me out with their abilities."

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

AR: Augmented reality

GPIO: General Purpose Input/output

LAN: Local Area Network

LED: Light Emitting Diode

USB: Universal Serial Bus

RFID: Radio Frequency Identification

RGB: Red, Green and Blue

RQ: Review Question

RoboToy: Augmented Reality in A Shooting Game with Real Robots

WIFI: Wireless Fidelity

CHAPTER…1:
INTRODUCTION

1.1 Problem Statement

Robotics is a broad field and everyday there is a pioneering invention in the field. Robots were invented by the humans just for fun but by now they are used for assisting humans in various sectors. Human beings are better suitable for multifaceted, imaginative, adaptive jobs, and robots are good for dreary, recurring tasks, permitting human beings to do the harder thinking jobs, whereas a robot is employed for substituting humans for various recurring tasks or entertainment to make living more expedient. To make living more expedient we are required to develop augment reality in a shooting game with real robots.

1.2 Aims and Objectives

It's is a game with real robots that you can control and watch through a mobile device or a personal computer. You can think about it as some kind of 'first person shooter' style of game with real world graphics. In the place of some 3d rendered scene you will see video streamed by a mounted camera on your real-world robot. Whenever you press the movement buttons, your robot will respond accordingly. Whenever you 'shoot' your 'cannon' using on screen game pad, a LED cannon mounted on your robot will flash and hopefully will hit an amplified light detector mounted on another robot. Player controllers and robots communicates with each other through an existent LAN or through an Access Point setup in one of the robots. Basically, this is the general game idea.

1.3 Background Information

This segment is deliberated with a summary of the history of robotics. As you may have presumed, the robotics history is entangled with the history of science, technology and the fundamental principles of progress. Technology employed in electronics, computers, even pneumatics & hydraulics can all be measured as a fraction of the robotics history. Robotics at present symbolizes one of the mankind's supreme achievements and is the only best endeavor of mankind to create an artificial, electronic being.

1.3.1 Before 20th century

- Though the division of Robotics is new, the making of Robots initiated in the year 1250 when the first man-made automated human (Robot) was developed. In the phase from 1250 to 1950 the Robots were created for entertaining rather than for applications. (Agarwal)

1.3.2 During 20th century

- In the year 1921, the Czech dramatist "Karel Capek" coins the world by using the word robot in his play Rossum's Universal Robots (R.U.R). This word robot is derived from a Czech word which means "compulsory labor."
- "Runaround" was composed by Asimov about robots in the year 1942, it held the "Three rules for robots"
- Robots are not harmful to the humans, or through working, permit a human to come and damage.
- A robot must follow the commands given by human beings apart from where such instructions would conflict with the First Law of Robotics.
- A robot must defend its own survival providing such safety does not clash with the First and the Second Law of Robotics.
- In the year 1956, George Devol and Joseph Engelberger established the first robot company.
- In the year 1959, computer assisted manufacturing was verified at MIT. (Agarwal)
- UNIMATE- The first industrialized robot was online in a General Motors automobile plant, in the year 1961.
- 1963 was a revolutionary year, first computer controlled robotic arm was designed and it was named as Rancho Arm. The invention was basically for the handicapped peoples.

The inventions in the field of Robotics were never ending and gave human beings a sudden surprising gift as & when launched. After Rancho's Arm various other inventions too were done, but all the above was the first among all. (Agarwal)

1.4 Deliverables

- It's a game that you can play with your mobile device or a computer, being on the control of real robots.
- This game is supposed to be played with 2 or more robots, one player controlling each one.
- Each robot will work as a web application server running all the game logic and will synchronize game status with each other robot.
- 3D video games could increase memory capacity
- Gaming could be good for pain relief
- Some research shows that video games might make you smarter.

1.5 Tools used

The Augmented Reality in A Shooting Game with Real Robots is powered by:

- Raspberry Pi 3 B
- Arduino Uno r3
- Raspberry Pi Camera (model 1.3 or 2)
- 3D printed parts (e.g. the 'turret', see wiki)
- DC motors
- Motor Shield (H bridge)
- LEDs, sensors and some other components
- battery (USB power bank)
- optional USB WIFI dongle
- MFRC522 RFID module

CHAPTER…2:
LITERATURE REVIEW

This section presents a literature review of previous studies of Augmented Reality (AR) games for learning. We classified learner groups, learning subjects, and learning environments mentioned in the literature. From this we conclude that AR games for learning generally have positive effects. We found that the most reported effects for AR learning games were the enhancement of learning performance and the learning experience in terms of fun, interest, and enjoyment. The most commonly used measurements for learning achievements were pre-test and post-test regarding knowledge content, while observations, questionnaires, and interviews were all frequently used to determine motivation. We also found that social interactions were encouraged by AR learning games, especially collaboration among students. The most commonly used game elements included quizzes and goal-setting. Extra instructional materials, 3D models, and face-to-face interactions were most frequently used for AR features. In addition, we came up with five suggestions for the design of AR learning games based on reviewed studies. In conclusion, six interesting findings were discussed in detail in the review, and suggestions for future study were offered to fill the research gaps.

2.1 Introduction

Augmented Reality (AR) is the technology that overlaps virtual objects onto the real-world objects. It has three key features: the combination of the real world and the virtual world, real-time interaction, and 3D registration. The past few years have witnessed a growing popularity in the research interest for AR since mobile devices such as smartphones and tablets have offered much easier and cheaper access to AR for users than before. Positive effects of AR technology on students' learning were identified in previous studies in the development of skills and knowledge, enhancement of learning experiences, and improvement of collaborative learning. The use of AR in education could improve the learning efficiency and provide a more fun experience for students.

Serious games can be defined as computer games with educational purposes and see entertainment as an added value. Serious games are gaining increasing importance in education, providing an enhanced experience in learning. They were found to be effective with respect to learning and retention. Other frequently reported outcomes included knowledge acquisition and motivational outcomes. AR games refer to the digital games that are played in a real-world environment with a virtual layer on top of it. It is possible for players to interact with both the objects in the virtual world and people in the real world, avoiding the social isolation. With the advantages and positive outcomes of AR technology and serious games in the educational field, a growing number of studies focusing on AR games for learning have emerged in the past few years (e.g. studies of Eco Mobile and study of Mad City Mystery).

2.1.1 Features of AR

The features of AR lead to a variety of positive effects on learning. The interactive 3D models in AR can enhance students' learning experience and collaborative skills; the combination of the real world and the virtual world in AR can support the study of the invisible concept and content; and the rich instructional materials (e.g., text, video, audio, etc.) can attract and immerse students into the learning. In addition to that, some literatures drew attention to the social impacts of AR on students. For instance, the use of AR technology provided more opportunities for students to communicate and collaborate in the real world. The social interactions between students and teachers, students and their parents were also encouraged. Structured literature reviews were found on AR for educational purposes. For example, one systematic review of AR for education investigated 68 AR studies in education and concluded several advantages and challenges. In this review, the advantages of AR in educational settings were classified into learner outcomes, pedagogical contributions, interaction, and others. In another review, the definitions, taxonomies, and technologies of AR technology were introduced, and the AR features and their affordances, as well as the solutions for AR challenges, were discovered. Different affordances of location-based AR and image-based AR for science learning were also studied. The review on AR trends in education found the educational field and purposes, target group, advantages, data collection methods, and discussed the trends for AR for educational settings. These literature reviews mainly focused on the advantages and affordances of AR. Moreover, the purposes of using AR were different, such as a practice for lab experiments or an introduction to certain topics. However, the game elements and social factors contained in the AR applications were not addressed enough.

2.1.2 Purpose of the Study

As can be seen above, numerous studies have been done on the use of AR technology or games in education for students. However, the efficacy of AR learning games as an integrated concept is less well known, let alone what would be successful design strategies for AR learning games. Therefore, the aim of this study is to present a systematic literature review on AR learning games, considering the current state of AR learning games, their effects on students regarding learning outcomes and social interactions, their evaluation techniques, as well as their design principles. To achieve this aim, this study identified and analyzed 26 research articles with the educational use of AR games, published from 2006 to 2016.

2.2 Method

2.2.1 Selection Process

In this study, we first searched via Google Scholar. The search terms we used were "augmented reality" combined with "serious games", "learning and games", and "education and games". Additionally, we investigated the references of previous reviews on AR technology in educational field to find relevant studies. We found 63 studies first. Then we examined the selected studies using a set of inclusion and exclusion criteria (see Table 2.1) and determined whether they were related to the purpose of this study. After the examination, 26 articles were found to be highly relevant to the purpose of this study.

Table 2. 1 : Inclusion and exclusion criteria

Inclusion criteria	Exclusion criteria
Involve AR as primary component	No clear data collection method
Involve game play in the design	Only introduction without evaluation
For educational purposes	For target groups with special needs

2.2.2 The Data Coding and Analysis Process

The first research question addresses the learner groups, subjects, and environments of AR learning games. The learner groups were divided into kindergarten, primary school students, middle school students, high school students, college students, and not specified in the article. In some studies, more than one learner group are used, then more than one category was applied for the learner group. The subjects of AR learning games were divided by looking for subject-related words in the article. One study might also apply to more than one code for subjects. The environments of using AR learning games were divided into five categories, which were: outdoors, classroom, home, others, and no limits. The effects and their measurements of AR learning games were coded by reading through the data collection, method, findings, results, discussions, and conclusion sections from the 26 articles. We looked for coding words to identify the effects and measurements. For effects, we used two main categories, learning achievement and motivation. Learning achievement was related to the learning performance, learning effectiveness, and the cognitive load of the knowledge content, while motivation related to a broad view including engagement, interests, fun, satisfaction, and positive attitudes. It should be noted that in some studies, more than one effect could be found, so more than one code might be applied. As for the effects of social interaction in AR learning games, we looked up the coding words: social, collaboration, competition, guide, discussion, communication, reflection, and share from the 26 articles. To code the elements/features used in AR learning games, we read through the design, implementation, and procedure sections from the 26 articles, searching for keywords from the description of the AR games. Sub-categories were classified into two main categories, AR features and game elements.

After the review of previous studies of AR learning games, we have six interesting findings. First, the subjects and learning content used in previous studies were too narrow. Studies for Science & Biology attracted most of the attention while there were fewer studies focused on the other subjects, such as Literacy and Mathematics. Secondly, most of the current AR learning games was played outdoors or in the classrooms. However, since students spend a lot of time at home and play digital games, it might be more effective to design AR learning games that can be played at home, which may encourage them to study spontaneously and in a more fun way. Thirdly, a notable gap was found in the retention effects. As for the measurements in previous studies, some commonly used instruments were addressed, while some studies didn't mention how they created and evaluated their instruments. More attention should be paid to the proper measurements for the effects. Fifth, social interaction effects were found by playing AR learning games, especially among students. However, little research focused on how these social interactions affected the learning achievement or motivation in turn. Also, more social effects were found among students than between student and teacher, or student and parents. The AR games that focus more on the interactions between student and teacher, or student and parents may lead to beneficial results for both sides. Finally, we found various game elements and AR features were used in the design of the AR learning games. However, there is a lack of systematic research on how different AR features and game elements influenced or supported the effects specifically.

CHAPTER…3:

DESIGN METHODOLOGIES

3.1 Packaging

You should first make sure that you have installed all the required packages in Raspbian before installing Robotoy. This means you should get all these packages installed and working (look at Wiki sections for more detailed information about these procedures):

- Java JDK 1.8
- Raspberry Pi Camera
- UV4L + WebRTC
- Wiring + Pi4J

RXTXcomm (unless you are not using Arduino with Raspberry Pi for controlling motors)

These packages are optional for running Robotoy, but are highly recommended:

- HostAPD (make it possible to turn your Robot into a Wireless Access Point)
- DNS Masq (usually it's used with host pad for providing DHCP and DNS support)
- Avahi Daemon (make it easy to find you robot in your local network)

Note: I've only tested it with Raspbian OS so far.

Of course, you should also have a network interface on your Raspberry up and running and have an SSH terminal for doing all the configurations. It's recommended that you do this wirelessly, since you probably don't want cables attached to your robot once it starts moving around.

3.2 Assembling

The 'hardware' part of Robotoy should also be assembled before installing the 'software' part. There may be different robot versions with various parts and with the proper adjustments you could make use of the same source code. These instructions apply to the default 'robot setup': the robot model 'Alpha'. 'Alpha' robot model hardware parts are shown in Table 3.1

Table 3. 2 : Product List

Serial No.	Product Name
`	Raspberry Pi 3 'B'
2	Raspberry Camera Module (v1.3 or newer)
3	Arduino Uno r3 (DIP)
4	Arduino Motor Driver Shield (tested with Ada fruit Motor Driver Shield version 1)
5	Kit of Car Chassis + DC motors + wheels (tested with a common 4WD car chassis kit)
6	Power Bank 10,000mAh with two USB ports
7	MFRC522 RFID module kit (13.56MHz) including some RFID cards
8	3D printed turret (PLA FDM).
9	USB cable for powering Raspberry (Type A plug to Micro Type B plug)
10	USB cable for connecting Arduino to Raspberry (Type A plug to Type B plug)
11	USB cable for powering motors (Type A plug to TTL, or cut one end off)
12	Micro SD (suggested a class 4 with 16 GB)
13	Led IR TSAL 6200
14	Led any visible color 5mm (e.g. yellow)
15	Led RGB 5mm common anode (also works with a LED with common cathode if you change some settings later)
16	BC547 NPN transistor
17	TSOP 4838 IR light detectors
18	Swivel ball caster (if you want to build 2WD robot instead of a 4WD robot)
19	Dupont cables package (male-female) plus (female-female)
20	Mini breadboard (170 points)
21	Resistors (1 x 10 KOhm, 2 x 220 Ohm, 1 x 100 Ohm, 2 x 47 Ohm)
22	screws + hex nuts M2.5 10mm (8 pieces)

23	screws + hex nuts M3 10mm (10 pieces)
24	metal angle brackets (90 degrees with each leg 3.5cm long, 2mm thick, holes about 7 mm diameter)
25	spacer rings

From this point on, we assume all the required external packages have been already installed. This project is made of some software parts and some hardware parts.

3.3 Timeline Diagram

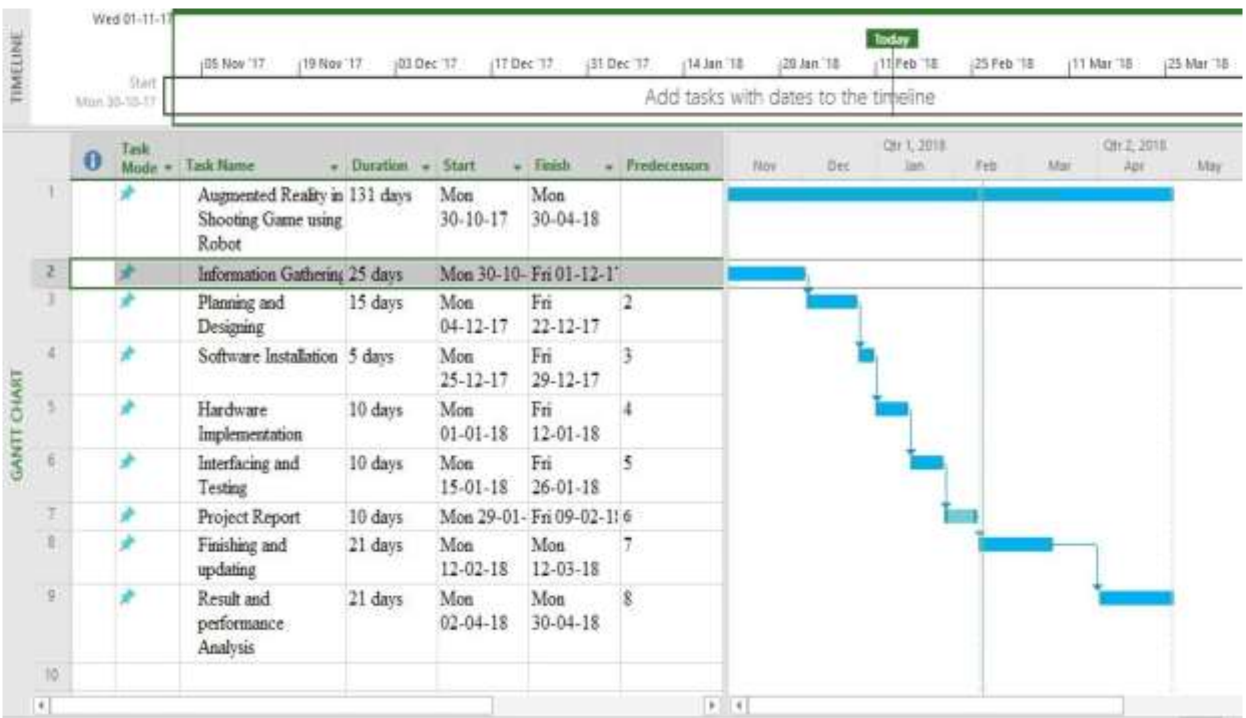

Figure 3. 1 : Timeline Gantt Chart

3.4 Block Diagrams

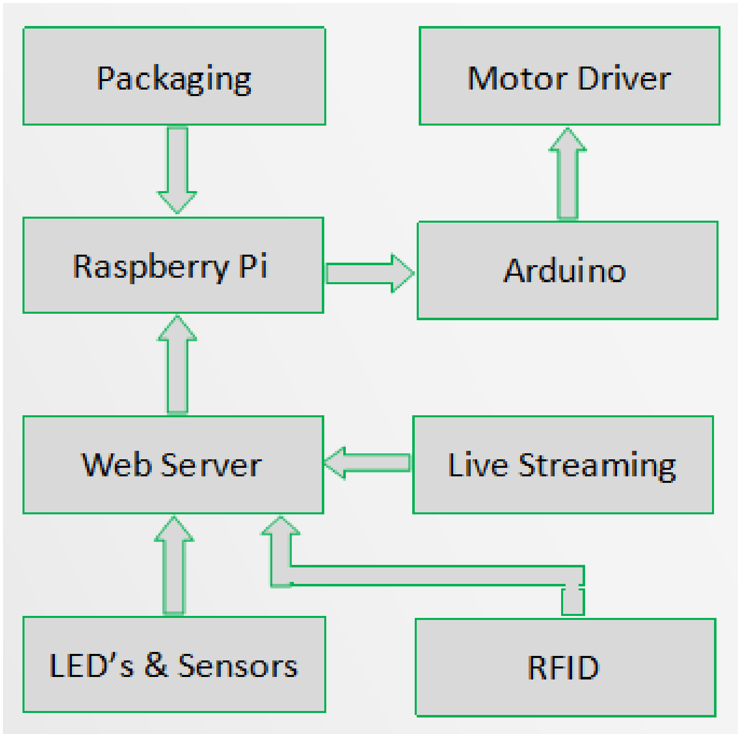

Figure 3. 2 : Hardware Design

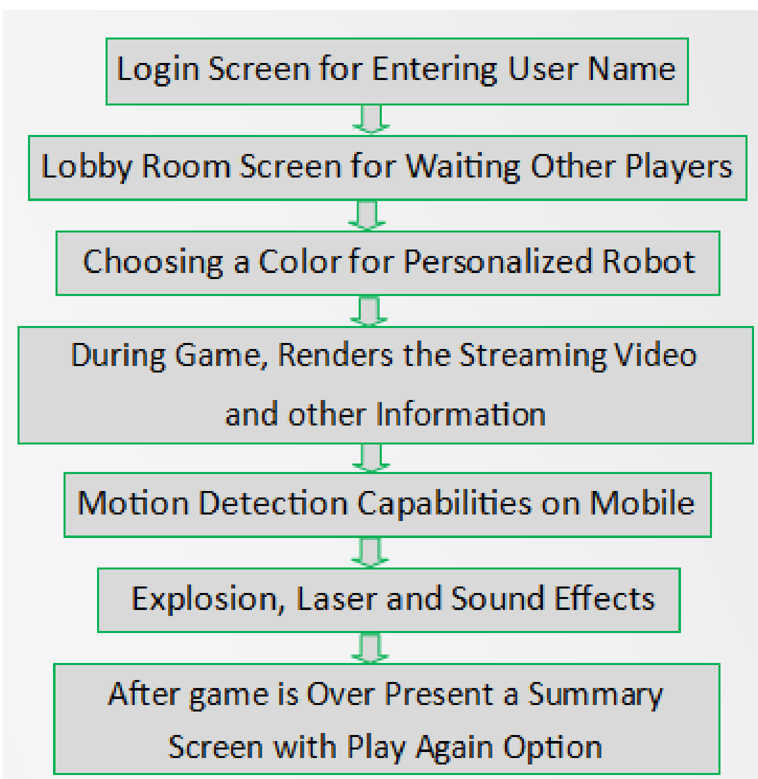

Figure 3. 3 : Software design

3.5 Software Parts

The Robotoy software is powered by these third-party technologies:

- Java
- HTML5
- JavaScript
- CSS
- JSP
- JSP Expression Language (EL)
- JSTL
- jQuery
- Data Tables (js)
- Arduino C language

3.6 External Dependencies

External Robotoy Java Library dependencies (note: these dependencies are automatically resolved with the 'Gradle' build):

- Jetty (http://www.eclipse.org/jetty/)
- Bouncy Castle (http://www.bouncycastle.org/)
- GSON - Google JSON (http://github.com/google/gson)
- Commons Lang Apache (http://commons.apache.org/proper/commons-lang/)

External Robotoy dependencies:

- Wiring + Pi4J (http://pi4j.com/)
- UV4L + WebRTC (http://www.linux-projects.org/uv4l/)
- RXTX Serial Communication (http://rxtx.qbang.org/)
- SPI bus (Serial Peripheral Interface)

3.7 Hardware Parts

The (suggested) Robotoy hardware (Alpha robot model) is powered by:

- Raspberry Pi 3 B
- Arduino Uno r3

- Raspberry Pi Camera (model 1.3 or 2)
- 3D printed parts (e.g. the 'turret', see wiki)
- DC motors
- Motor Shield (H bridge)
- LEDs, sensors and some other components
- battery (USB power bank)
- optional USB WIFI dongle
- MFRC522 RFID module

3.8 Web Server

Robotoy uses an embedded JETTY web server for distribution of its graphical user interface and provides 'web-sockets' communication between robots and between players. There is no need for an external server. Each robot is a server on its own right and can synchronize its information with others. These are the features implemented in this module:

- Listen to HTTP and HTTPS ports and provides a web context with all resources embedded in the application (JSP, JS, TLD, CSS, MP3, PNG).
- Listen to web sockets calls and keep a pool of connections for synchronization of game internal state among players and robots.
- Provides a simple RESTful interface for debugging purpose.

3.9 Auto-Discovery Engine

Each Robotoy can recognize other Robotoy in the same LAN automatically. There is no need to pre-program a list of available Robotoy. You can add more Robotoy to your game as needed but be aware that each one will be streaming video on the same LAN, increasing network traffic. I guess that 8 robots should be enough (you need at least 2 robots). If you want more robots, you will have to tweak some internal parts of the code. These are the features implemented in this module:

- Broadcasts to the local network a multicast message informing about the robot existence.
- Acknowledges other robot's broadcast messages.

3.10 Motor Driver

This module controls the robot movements. These are the features implemented in this module:

- Control robot's DC motors using different approaches:

a) Delegating the motor control to an Arduino board, which in turn have a serial communication via USB.

b) Direct wiring between Raspberry's GPIO and an external motor driver board.

- Control robot's speed using PWM (either hardware or software depending on the robot configuration).

3.11 Optical Circuit

This module is related to the weaponry and shield of each robot. It uses infrared signals with pulse width modulation and a simple protocol to make this game behave like a 'laser tag' game. These are the features implemented in this module:

- IR with pulse width modulation is used for communication between robot that fires a laser beam and the robot that detects it.
- Additional RGB led used for easy robot identification (it's not part of the targeting system).

3.12 Game Engine

Several rules related to the game are managed by each robot and the overall game state is synchronized among the robots. These are the features implemented in this module:

- Holds the full game logic about players, robots and goals.
- Broadcasts all the game logic between participants.

3.13 Graphical User Interface

This module includes graphics, sound and some special effects displayed during gameplay. The game can be played by any device that supports HTML5 standard, including smartphones, tablets and computers with not-too-old browsers. Tested with:

- Google Chrome (version 54)
- Safari on iPhone 6 (iOS 10)
- Safari on iPad 2 (iOS 10)

These are the features implemented in this module:

- Login screen for entering a user name.
- Lobby room screen for waiting other players to join and take control of available robots and choosing a color for identifying the robot.
- While the game is in the 'PLAY' stage, renders the streaming video image and over it renders some additional information, such as the total 'in game' life remaining, robot's WIFI signal strength, gamepad buttons, etc.
- Motion detection capabilities on the controlling device (e.g. accelerometer and gyroscope) may be used to control the robot, just like an analogic joystick would do (e.g. make it go forward, backward and spinning around).
- Explosions and laser effects are rendered on the display above the streaming video image.
- Play sound effects.
- After the game is over, presents a summary screen with stats.
- Additional administration pages for debugging and for setup of a new WIFI network.

3.14 RFID

This module is related to RFID cards and sensors. Each Robotoy is equipped with a RFID module working at 13.56 MHz attached to its base. It's used for detecting some RFID cards that are spread over the playfield. The game will treat these RFID cards in a unique way, usually making the robot power up (i.e. restoring life points that it may have lost in a previous fight). These are the features implemented in this module:

- Scanning for RFID signals that are reflected from RFID tags.
- Keep and synchronizes status about each RFID tag found during gameplay.

CHAPTER…4:
HARDWARE IMPLEMENTATIONS/SOFTWARE SIMULATIONS

Augment reality in a shooting game with real robots that you can control and watch through a mobile device or a personal computer. You can think about it as some kind of 'first person shooter' style of game with real world graphics. In the place of some 3d rendered scene you will see video streamed by a mounted camera on your real-world robot. Whenever you press the movement buttons, your robot will respond accordingly. Whenever you 'shoot' your 'cannon' using on screen game pad, a LED cannon mounted on your robot will flash and hopefully will hit an amplified light detector mounted on another robot. Player controllers and robots communicates with each other through an existent LAN or through an Access Point setup in one of the robots. Basically, this is the general game idea. You will be using a web browser in your mobile device or computer to play this game. It should be compatible with HTML5 and should not be too old. Have been tested with latest versions of Chrome and Safari. Each robot will work as a web application server running all the game logic and will synchronize game status with each other robot.

4.1 Robot Parts Lists

The parts list needed for building one robot is given in the Table 3.1. It's possible to use some different hardware configurations with minor software settings changes. For example, it's possible to connect the Raspberry GPIO directly to the Arduino motor Driver Shield without using Arduino. You may also try some other motor driver PCB pluggable directly over Raspberry Pi (such as Adafruit DC Motor HAT for Raspberry).

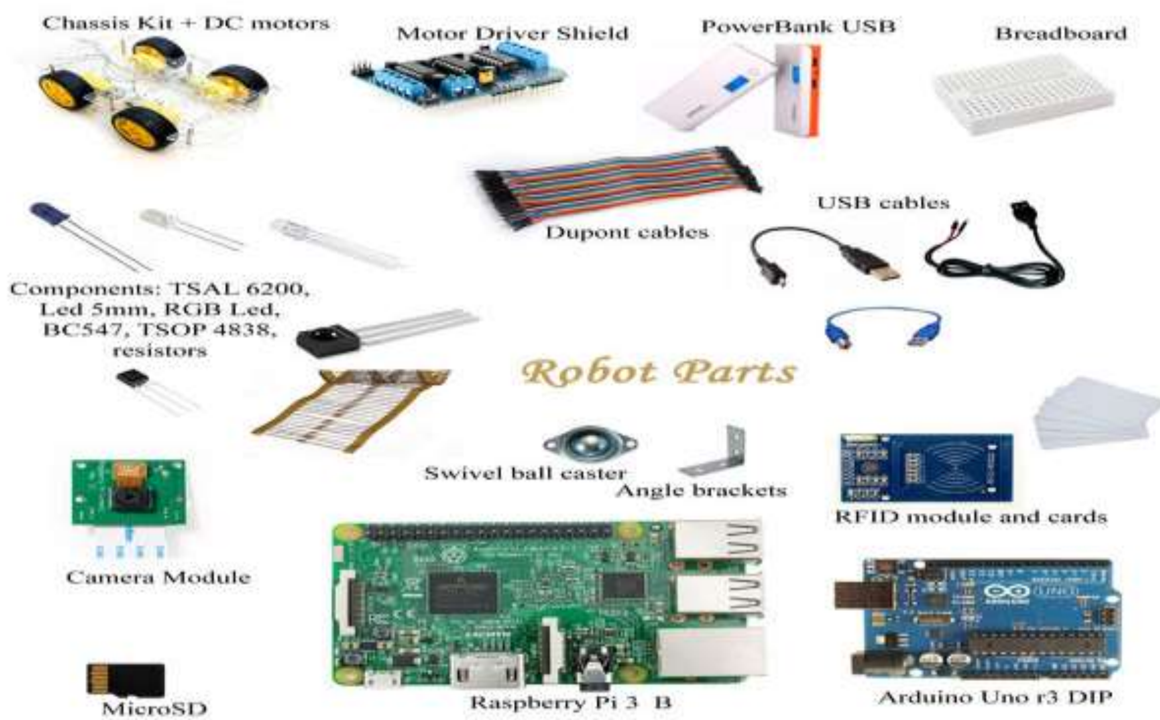

Figure 4. 4 : Components

4.2 Robot Lower Plate Assembly

In this step you are going to mount the DC motors and wheels on your robot chassis plus three circuit boards (Arduino, motor driver shields and MFRC522).

I built this robot with a very common and cheap kit that includes four sets of wheels and motors, two acrylic plates, screws and some other mounting parts. Please note that in my experience with this kit I found it difficult to make the turning movements with all four wheels assembled, so I decided to keep only two wheels in my robot and I added a swivel ball caster to the front and I positioned the battery at robot's back to balance weight. The robot was spinning very easily after doing this. Feel free to use other arrangements. The Arduino board was positioned close to the motors, just ahead of it, with the USB connector pointing backwards. On top of Arduino was attached the motor driver shield.

The motor driver shield has a power jumper that indicates if there is a single DC power supply for both Arduino and motors. In this case it's better to use an external power supply for the motors, so the power jumper must be removed. Since we are using in this project a power cell with USB connectors, I took a regular USB cable and I cut one end off, exposing its internal wires. The VCC and GND wires are usually colored in red and black respectively but use a multimeter to make sure. These wires are connected to the external power header on the motor driver shield. The other USB wires (denoted D+ and D-) are not needed in this project.

The motor shield can drive up to four DC motors. They are labelled M1, M2, M3 and M4. In my project with 2 motorized wheels I wired the motors to the M3 and M4 connectors (two wires into each one). There are some configuration settings that can be adjusted later in another step-in order to use other connectors. The MFRC522 RFID module was positioned just ahead of Arduino on the acrylic plate. With this kit this leaves about 2cm space between the module's antenna and the floor, what should be enough for detecting a RFID card on the floor.

Thus, this is the final configuration of the lower acrylic plate: two DC motors mounted on the rear side, one swivel ball caster attached to the front side, a MFRC522 module attached close to it and an Arduino plus motor driver shield on the central part. Everything should be firmly attached to the acrylic plate using screws, spacers and hex nuts.

Figure 4. 5 : Chassis

4.3 Arduino Program

In this project the Arduino will be controlling the motor driver shield directly on demand of Raspberry Pi's logic. We use a serial communication between Raspberry and Arduino using the USB connection and a protocol specific for this project.

This protocol basically defines some short commands (like 'f' to command it to move forward and 'b' to move backwards). If you choose different motor driver numbers for your DC motors, you will have to change these variables in this program accordingly:

- #define WHEELS TWO_REAR
- #define REAR_LEFT 4
- #define REAR_RIGHT 3
- #define FRONT_LEFT 1
- #define FRONT_RIGHT 2

The first variable tells that we are going to control just two motors located behind the robot. Each number on the following variables refers to the motor number used on the motor driver shield. So, out-of-the-box this sketch file tells that each robot will have just two rear motors and we are wiring the rear left motor with the driver number 4 (labelled M4 on the board) and the rear right motor with the driver number 3 (labelled M3 on the board). You may edit this file using any text editor of your choice or preferably using the Arduino IDE.

Use the Arduino IDE to upload this sketch file into your Arduino using your computer and the USB cable connected to it.

4.4 Turret - Part 1

The turret comprehends the optical components of this robot. It includes the following parts:

- 1 RGB led
- 4 TSOP IR detectors (TSOP 4838)
- 1 IR led (TSAL 6200)
- 1 visible light led (e.g. yellow light with high brightness)
- resistors and transistor
- Raspberry camera

Before you proceed, check the datasheets for your components to identify the correct pin order. The TSOP 4838 I used here have the following pin order from left to right when the 'bump' is facing toward you: DATA, GND, VCC. The RGB led with common anode I'm using here have the following pin order from left to right when the anode (longer lead) is closer to the left side: RED, ANODE (+), GREEN, BLUE. For the remaining LEDs, just to remember, the longer lead is the positive (anode). If they have leads with the same length, look at the metal plate inside the LED. The smaller plate indicates the positive (anode) lead.

This project results in three 3d printed parts. Before gluing these parts together, connect the electronic components and join the wires. Do as follows:

1. Insert he RGB led's leads into the four small holes in the middle of the top part. You must keep them separated from each other.
2. Alongside the borders of this mounting part you will see four series of three dots markings. Carefully plug the four TSOP IR detectors on each one of them. The bump on each detector should be facing outwards.
3. Now it's time to fix cables to each of component leads. Choose distinct colors for each different purpose (e.g. black for GND, orange for VCC, yellow for data, gray for common RGB anode, red, green and blue for the corresponding RGB leads). If you have good soldering skills, try to solder one wire to each lead inside the turret. Otherwise, connect each lead to a dopant cable and use insulating tapes to stick each one of them. You may use a female-to-female dopant cable on the longer lead of RGB led (the common anode) and male-to-female cables for everything else. It's very important to use insulating tape to cover entirely all the leads because some of them will be touching the camera module. If you don't do this, you risk ruining your camera module.
4. Join two wires to the IR LED and another two to the visible light LED. Remember which wire color you choose for 'anode' and which one you choose for 'cathode' on each LED. Insert these LEDs through the turret's cannons.
5. The Raspberry camera is just about the right size to fit inside the turret 'head'. The square hole is the same size of the camera lens. You can hold it into the turret piece, leaving room for the wires of the cannon's LEDs just above it.
6. Glue the two parts forming the turret's 'head'.
7. Glue the turret's 'head' to its 'base'. Check the pictures below for the correct position.

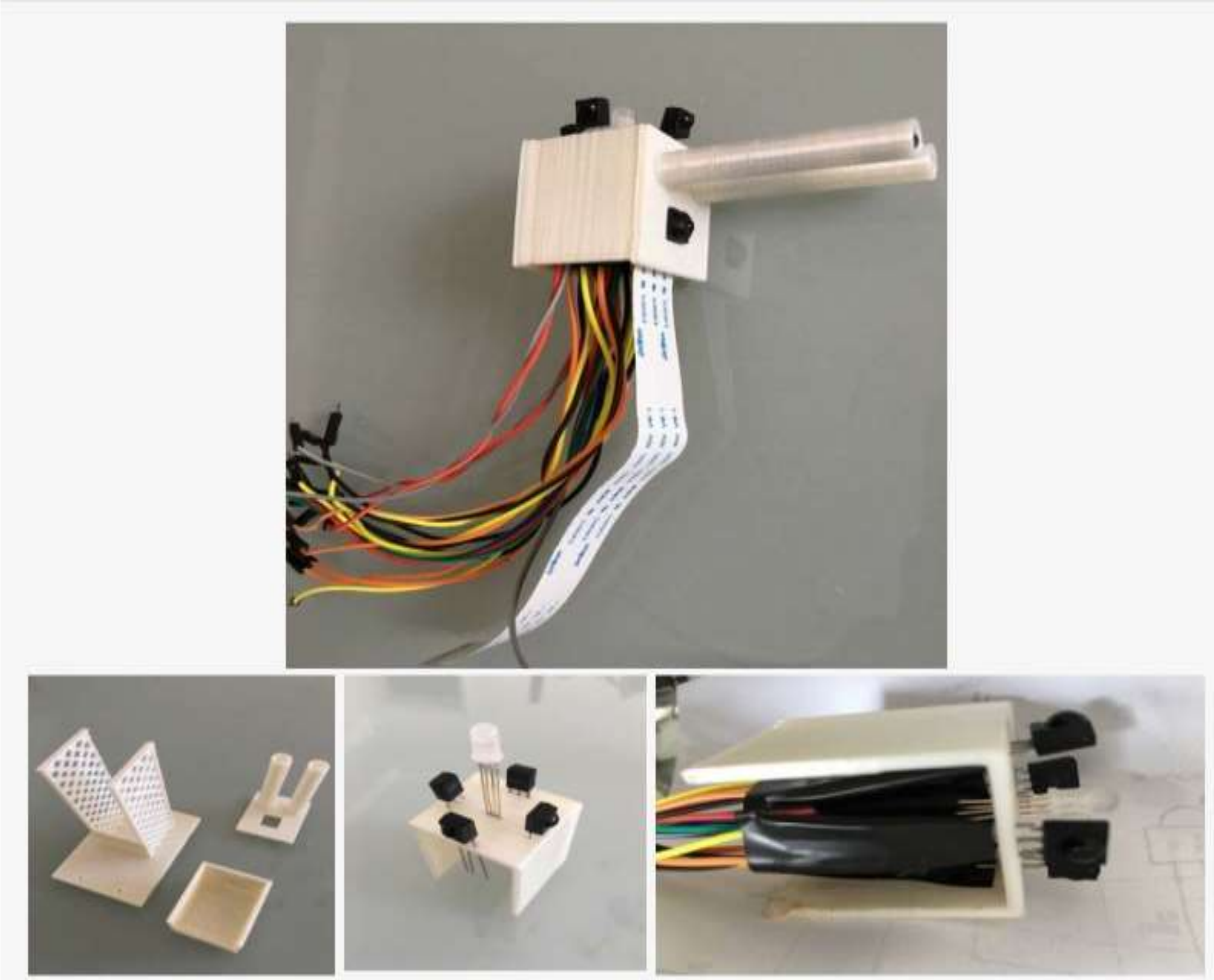

Figure 4. 6 : Turret

4.5 Turret - Part 2

The electronic circuit schematic is displayed below. You can either do this by soldering parts or by using dopant cables and a breadboard. In the pictures above, I'm using a solderless design. Basically, you will have:

- Three resistors with different values for the RGB led (one for each cathode). In this project I used 47 Ohms resistors for 'green' and 'blue' cathodes and 100 Ohms resistor for 'red' cathode. Check the datasheet for the LED you are using to choose proper resistance values. The current should be below 10 mA on each pin.
- 3.3V for the RGB led common anode (longer lead).
- A 10 kOhm resistor connected to the transistor base.

- Two 220 Ohms resistors connecting the transistor collector to the cannon's LED cathodes (they are connected in parallel).
- 5V for the cannon's LED anodes and for the TSOP detectors.
- GND for the transistor emitter and for the TSOP detectors.

Note: the TSOP used in this circuitry are working at the 5 V level. To prevent it from damaging the GPIO of Raspberry, which should operate under 3.3 V level, we are going to use a voltage divider. According to the datasheet of TSOP 4838, they already have a 33 kOhm internal resistor between Vcc and 'data' pins. Using the internal 50 kOhm pull down resistors present in Raspberry Pi GPIO we would get the TSOP 'data' level down to a safe voltage level.

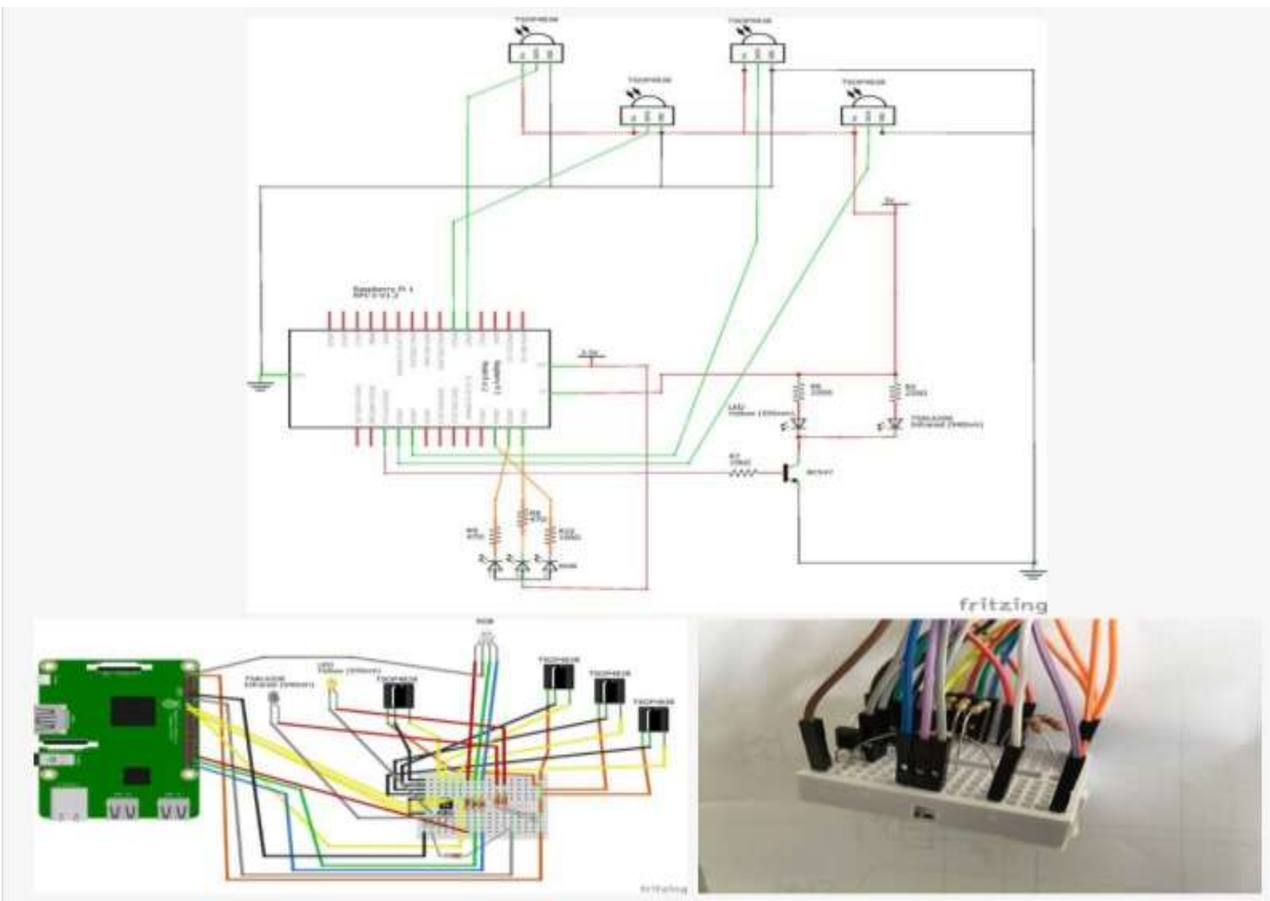

Figure 4. 7 : Breadboard Connections

4.6 Robot Upper Plate Assembly

On the upper acrylic plate, we are going to attach the remaining parts:

- Raspberry Pi on the front
- Turret base on the central part. Breadboard with the turret circuitry next to it.

- Battery (power cell) on the back
- Start by fixing the turret base on the central part of the top acrylic plate. Use screws through the holes on the 3d printed base.

In case you didn't do it yet, format the microSD card in your computer and load it with the NOOBS installation manager. This project assumes you will be using the Raspbian operating system. In case you need help with NOOBS download and operation (see Appendices A).

Once you have formatted and stored the NOOBS image in the microSD card, plug it in the Raspberry Pi.

Put the Raspberry Pi on the front of it with the USB connectors facing to the front. Use M2.5 screws to fix it to the plate. The power bank should stay behind to keep the overall weight balanced over the rear wheels. Fixing it on the acrylic plate may be tricky due to its shape. I was able to lock it in a steady position using the battery pack box. Use two angle brackets to fix the battery pack box on the acrylic plate and one more to support the battery inside it. I also had to cut small rectangular holes on the upper side of the pack box to expose the battery USB connectors.

The breadboard containing the turret circuitry was positioned to its side. The camera module flat cable was connected to the Raspberry CSI port (remember the Raspberry USB connectors should be facing ahead to prevent connecting the camera the wrong way). Then all connections to the Raspberry GPIO were made according to the diagram above. Overall there will be:

- One 3.3 V power line going to the RGB led anode lead directly
- One 5 V power line going to the breadboard which in turn will power all four TSOPs and the two cannon LEDs.
- One GND line going to the breadboard which in turn will ground all four TSOPs and the transistor emitter lead used by the two cannon LEDs.
- Three GPIO used as 'output' to the RGB led cathodes (red, green and blue through resistors on the breadboard)
- Four GPIO used as 'input' from the four TSOP detectors.
- One GPIO used as 'output' to the transistor base through a 10 kOhm resistor (used for turning on the cannon LEDs).
- Seven pins to be connected to the RFID module attached on the lower acrylic plate.

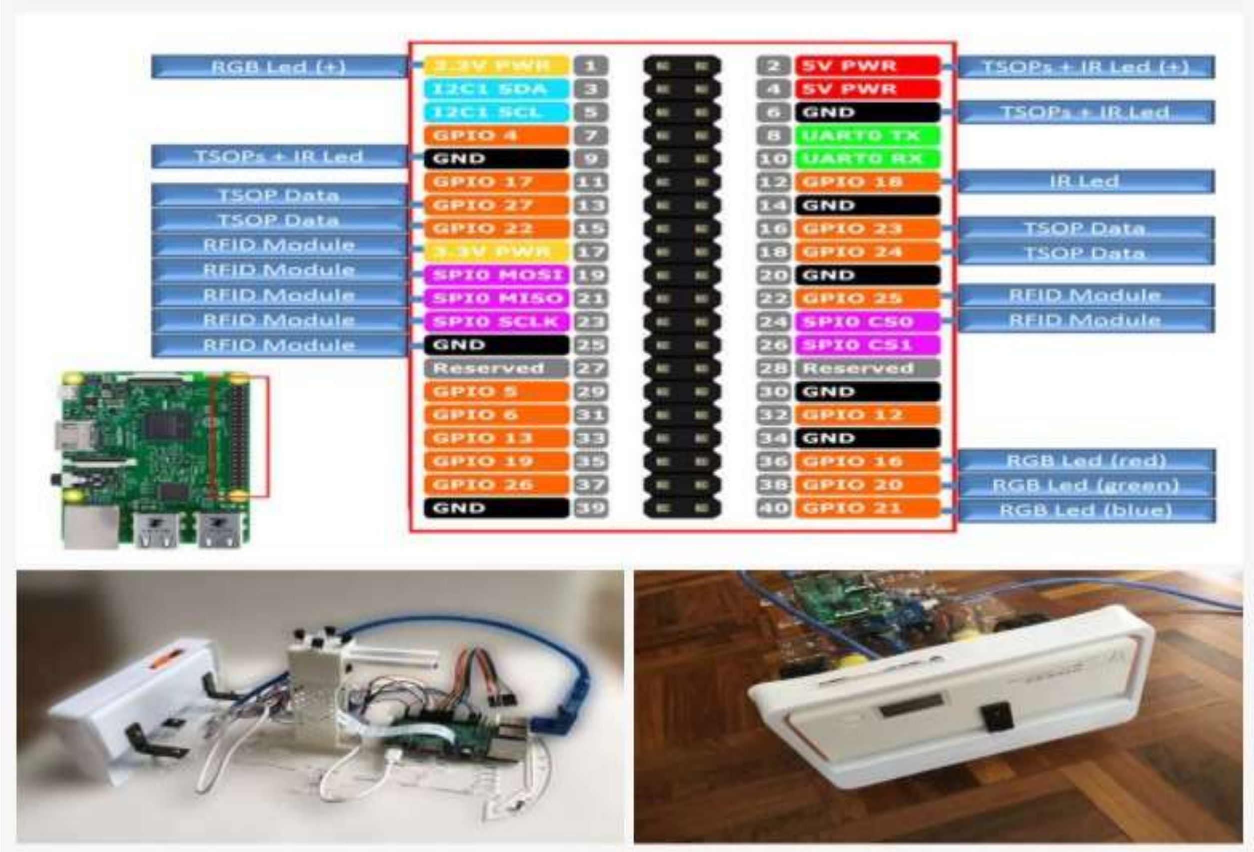

Figure 4. 8 : Upper Plate Assembly

4.7 Final Connections

Table 4. 1 : Connection List

Raspberry Pi pin number	MFRC522 pin position	Designation
w10 bcm08 p24	SDA	SPI Channel
w14 bcm11 p23	SCK	SPI Clock
w12 bcm10 p19	MOSI	SPI MOSI (Master Out)
w13 bcm09 p21	MISO	SPI MISO (Master In)
none	IRQ	IRQ (not used)
p25	GND	Ground
w06 bcm25 p22	RST	Not Reset and Power-down
p17	3.3V	3.3V

We are almost done with the hardware part of this project.

- Attaches the six-cylinder screw columns over the lower plate (were lies the Arduino board and all the motors). Use three on each side of the chassis.
- Connect the Raspberry Pi on the upper plate to the MFRC522 RFID module on the lower plate using seven wires as shown on the table above.
- Take a USB cable to connect to the Arduino board mounted on the lower plate and pass it over the small round hole on the upper plate.
- Attaches the upper plate over the lower plate using six screws with the six separating cylinders. Plug the USB cable coming from the Arduino into one of the USB ports of Raspberry. Use the 'L' angled USB connector as needed.
- Plug the USB cable connected to the 'external power' connection of the motor driver to the 2A USB port on the battery.
- Connect the universal serial bus cable to the micro-USB port on Raspberry-Pi and the other end to the other USB port of the battery.
- Turn on the battery and hopefully everything will turn on nicely.

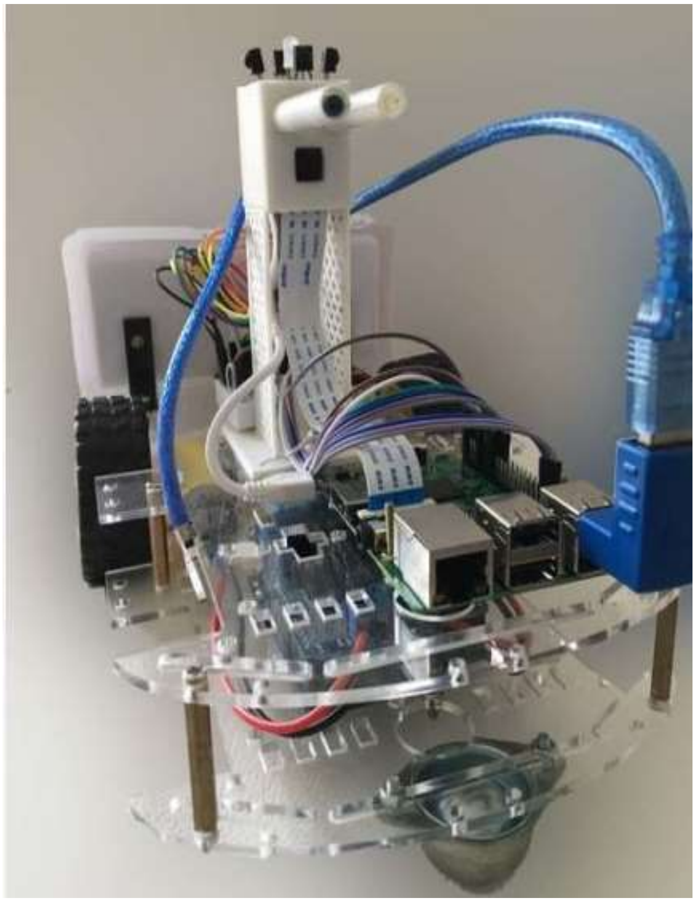

Figure 4. 9 : Final Connections

4.8 Raspbian Setup

This project assumes you will be using Raspbian operating system in your Raspberry Pi. There are some software parts needed by this project:

- Raspbian operating system
- Java JDK 1.8
- UV4L + WebRTC
- Wiring + Pi4J
- RXTX
- SPI bus and camera module enabled

If you did not have time to prepare the Raspberry Pi for SSH session using Wi-Fi, you will need to plug a monitor to the Raspberry Pi using its HDMI connector and plug mouse and keyboard using its extra USB ports.

It's important to make the Raspberry accessible through Wi-Fi because you won't like your monitor and peripherals hanging over your robot once it starts moving around. The SSH session will be used for configuration and maintenance, but you can replace it with a VNC session if you prefer a graphical interface.

4.9 Robotoy Program

Finally, you must install and configure the program that will control everything on the robot and will service the game pages for your controlling devices.

I already shared the whole application including an embedded application server and all the game logic. It's called 'Robotoy' and its totally open source free software (see Appendices B).

CHAPTER…5:
RESULTS AND DISCUSSION

5.1 Learner Groups, Subjects, and Environments for AR Learning Games

Regarding the “learner group”, we found that in the past decade, most of the AR learning games focused on primary school students (31%) and middle school students (29%). High school students (20%) followed primary school students and middle school students in popularity (see Fig. 1). Two studies focused on college students (8%). Two studies designed the AR games for college students in the majors of Design and Physics. One potential explanation for this could be that AR learning games can have a positive influence on younger students because they are more evocative and align better to the kind of games they are playing at home. Regarding to the learning “subjects”, Science and Biology (38%) were highly focused subjects in the reviewed AR learning games. This might be because the AR technology can provide advantages in reflecting the concept of knowledge in the real-world environment, allowing students to observe the objects in real-time.

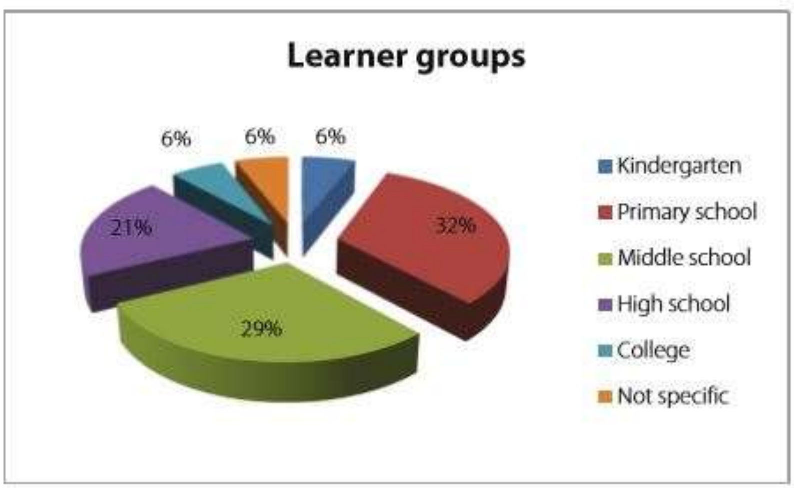

Figure 5. 10 : Learner groups

The study of Physics (12%), History (12%), and Art & Design (12%) were second preferred subjects. The real-time feature of AR enables students to receive feedback or see results immediately, which is favorable for subjects like Art & Design (see Table 5.1).

Table 5. 3 : Subjects of AR learning games

Subject	Number of papers	Percentage	Sample of research
Science & Biology	10	38%	[6]
Art & Design	3	12%	[2]
Physics	3	12%	[19]
History	3	12%	[32]
Mathematics	2	8%	[9]
Chemistry	2	8%	[33]
Literacy	2	8%	[15]
Others	3	12%	[32]
No specific	2	8%	[12]

We noticed a gap in the subjects of AR learning games. First, studies for Literacy (8%), Chemistry (8%) and Mathematics (8%) studying were underrepresented. The uneven situation of subjects should be broken down, which means more attention should be paid to a wider range of subjects. What's more, the existing studies for History learning often made a new game story for students to explore the history of a certain area. The location-based feature of AR technology enhances students' learning experience by allowing them to stand on the historical spots. Other subjects (12%) included culture study, 21st century skills, and library instruction study. However, we found little focus on the textbook-related content and knowledge, which is also valuable and should be paid attention to.

Regarding to the "environments" to use AR learning games, we saw significant preference from there viewed studies in using AR learning games outdoors (42%) and in the classrooms (35%). See Table 5.2. Playing outside is one of the advantages of AR learning games compared to other serious games, which may stimulate interest and excitement in students. On the other hand, it could be difficult for the teacher to control the learning process, and the safety issues should be considered as well. Students might come into dangerous situation, such as car accidents, when they put too much attention to their mobile devices. The AR learning games played in the classroom allowed students to play face-to-face and under the guidance of their teachers. Students could solve problems and collaborate with their classmates, and they could immediately get help and feedback from their teachers when they encountered problems or had questions (e.g. AR for preschoolers for Natural Science).

Table 5. 2 : Places to use AR learning games

Subject	Number of papers	Percentage	Sample of research
Classroom	11	42%	[24]
Outdoors	9	35%	[28]
No limits	4	15%	[22]
Home	1	4%	[21]
Others	1	4%	[8]

We found four AR games (15%) with no limits for the environment, and only one AR learning game (4%) was designed specifically for playing at home with the help of parents. Students, especially younger students, spend more time at home than in the classroom, and tend to spend a lot of time playing digital games. Meanwhile, their parents are curious about their learning status. Therefore, it can be effective to design AR learning games that students can play at home. It may encourage them to study more spontaneously and in a more fun way. In addition to that, their safety and communication with parents could also be addressed.

5.2 Effects of AR Learning Games on Students

Effects of AR learning games were classified into two main categories in this review, which were learning achievement and motivation (see Table 5.3). Regarding to the "learning achievement", half of the reviewed studies reported that AR learning games led to the effective outcomes in achieving learning gains in terms of learning content (e.g. AR for electromagnetism and AR system for library instruction). The positive effects also included the enhancement of learning efficiency (15%) and cognitive skills like problem-solving skills, critical thinking skills, multitasking skills and so on (12%). One study (4%) reported that the use of AR learning game could reduce the cognitive load of students. while on the contrary, another study showed that students felt frequently overloaded and confused due to the big amount of materials and tasks during the game play. The rest of the reviewed studies (19%) either found AR games were ineffective in the learning achievement or didn't focus on the learning achievement in their studies.

Table 5. 3 : The effects of AR learning games

Effects	Sub-categories	Numbers of papers	Percentage	Sample research
Learning achievement	Achieve learning gains	13	50%	[18]
	Enhance learning efficiency	4	15%	[10]
	Enhance cognitive skills	3	12%	[30]
	Decrease cognitive load	1	4%	[10]
Motivation	Enhance fun, interest, enjoyment	16	62%	[1]
	Enhance engagement	12	46%	[6]
	Enhance satisfaction	5	19%	[7]
	Enhance willingness to learn	5	19%	[22]
	Provide positive attitude	4	15%	[16]
	Enhance attention	4	15%	[25]
	Enhance confidence	4	15%	[22]

The motivation aspects involved engagement, satisfaction, fun, enjoyment, interest, attention, confidence and positive attitudes of students (e.g. System and AR-based educational game. Previous studies frequently reported that students described the learning experience with AR games as joyful and playful as they had fun playing AR games to learn school knowledge. We found similar result in our review that most of the students (62%) mentioned AR learning games as fun, interesting, or enjoyable (e.g. AR gaming in sustainable design education and mathematical education game based on AR). Nearly half of the studies (46%) also reported that AR learning games engaged them more than traditional learning methods (e.g. AR system for a visual art course). In addition to these two effects, AR learning games were also evaluated to "enhance satisfaction" (19%), "enhance the willingness to learn" (19%), "enhance attention" (15%), "enhance confidence" (15%), and "enhance positive learning attitude" (15%). Considerably less well studied were the retention effects. Nearly all studies tested the outcomes immediately after the use of the AR games.

In addition, most of the students never used AR games before, so a potential novelty effect of a modern technology might influence the research results. Therefore, more research should be done focusing on both short-term and long-term impacts on students after learning with AR games. Different measurements were used to evaluate the effects caused by AR learning games. used post-test in their experiments (e.g. AR for preschoolers for Natural Science). From the result we can see that only 50% studies measured learning achievement in terms of the knowledge content, indicating the rest of studies didn't use a proper test or didn't focus on learning at all. Regarding to the motivation aspect, most of the previous studies (65%) used observation as the main evaluation methods during students' learning and playing process (e.g. AR for enhancing library instruction).

Regarding to the motivation aspect, most of the previous studies (65%) used observation as the main evaluation methods during students' learning and playing process (e.g. AR for enhancing library instruction). The questionnaire also held a high popularity (58%) in the measurement of motivation (e.g. using AR games to teach 21st century skills). Some studies introduced and explained the questionnaire questions in their studies and Keller's ARCS Motivation Model was frequently adopted as the motivation questionnaire (e.g. AR system for a visual art course), whereas other studies didn't explain how they created and evaluated their questionnaire questions to measure the motivation accurately. Interviews to collect qualitative data was also widely used (42%). Pre-survey and post-survey (15%) were used to investigate the changes of attitudes before and after the use of AR games. See Table 5.4.

Table 5. 4 : Measurement methods of effects

Effect	Method	Number of papers	Percentage	Sample of research
Learning achievement	Pre-test and post-test	10	38%	[25]
	Post-test only	3	12%	[7]
Motivation	Observation	17	65%	[6]
	Questionnaire	15	58%	[33]
	Interview	11	42%	[11]
	Pre-survey and post-survey	4	15%	[30]

5.3 Effects of Social Interaction in AR Learning Games

Collaboration and interaction have emerged to be the main advantages of AR in education, since the technology allows users to work or study face-to-face in real life. Based on reviewed studies, we found three main types of social interactions, which were interactions among students, between teachers and students, and between students and parents. See Table 5.5. Most of the social interaction effects were found among students and the main effect was to encourage the collaboration (46%). In some AR games, students were required to work in groups to solve a certain task, while the competition (31%) among groups was also promoted. Evidence was also noted in the desire of sharing experiences with classmates (8%). Unlike the rich social interactions among students, the only social interaction between students & teacher (15%) and students & parents (8%) was guidance. Frequently, little attention was paid to the study of how these social interactions affected the learning achievement or motivation in turn. The attitude from classmates, the feedback from teachers, and the help from parents may all have an impact on children's learning outcomes. In addition, AR games should focus more on the interactions between student and teacher, student and parents.

Table 5. 5 : Social interactions

Type	Effect	Number of papers	Percentage	Sample of research
Student-student	Collaboration	12	46%	[5]
	Competition	8	31%	[18]
	Share	2	8%	[10]
Student-teacher	Guide	4	15%	[33]
Student-parents	Guide	2	8%	[15]

5.4 Elements or Features Used in AR Learning Games

AR learning games include AR features and game elements. Unique features or elements may have different outcomes regarding to learning achievement and motivation mentioned above. Therefore, this review sought to identify the frequently used AR features and game elements in AR learning games. Regarding to the "game elements", we found that time limitation was one of the most commonly used elements (46%) in the reviewed studies, which means students must finish the game in a certain period (e.g. AR science game). One reason to explain this might be because the attention span of students is limited, especially for younger students. Teachers might also find the time limitation helpful for them to control the learning progress, or researchers to control for variables in the experiment. The game elements of quiz-based (50%), inquiry-based (35%), and puzzles (30%) were also preferred by the reviewed studies. Students needed to answer questions or finish tasks in the game to continue. Game story (42%) was also another frequently included element in AR learning game design. Students started the game with story or background information, and some of them might play a role (27%) during the game.

Table 5. 6 : Game elements

	Game elements	Number of papers	Percentage	Sample of research
Game	Goals	13	50%	[25]
	Quiz-based	13	50%	[10]
	Time limitation	12	46%	[6]
	Game story	11	42%	[8]
	Inquiry-based	9	35%	[32]
	Collection game	8	30%	[5]
	Solve puzzles	8	30%	[13]
	Role play	7	27%	[32]
	Secret mission	4	15%	[18]
	Feedback	3	12%	[15]
	Board game	2	8%	[21]

Another frequently used game element was "collection" (30%). Players tried to look for different information and collect them to achieve the goals. The term "goals" was widely used in the reviewed literature, including the aims to get certain points, rewards, or finish a task (50%). Secret missions or hidden content were also included in some games (15%), the process of looking for the hidden mission might stimulate the interest of the students. The feedback element in the game design was mentioned three times (12%), and two board games (8%) were used. See Table 5.6. As for the AR features, 38% of reviewed studies used location-based AR, and the rest used image-based AR (62%). These two forms might have different advantages for learning. Since the visualization of knowledge content can promote to the fun experience of AR learning games for students, more than two third of the studies (77%) included extra instructional materials such as text, video, and audio. The 3D models were also used frequently in AR learning games (54%). Apart from that, some AR learning games (30%) also used physical objects, allowing students to interact in the game by using physical models. Communication in the real world is the main advantage of AR learning games (as opposed to regular videogames), and half of the reviewed studies encouraged face-to-face interactions in their games. AR presentation avatar (15%) and gesture-based input (12%) were also mentioned in previous studies. See Table 5.7.

Table 5. 7 : AR features

	AR features	Number of papers	Percentage	Sample of research
AR	Location based	10	38%	[25]
	Image based	16	62%	[8]
	Instructional material	20	77%	[5]
	3D model	14	54%	[15]
	Face-to-face	13	50%	[20]
	Physical model	8	30%	[16]
	AR presentation agent	4	15%	[13]
	Gesture-based input	3	12%	[14]

Although we have found the most commonly used game elements and AR features from previous studies, there is still a lack of research on how different AR features and game elements influenced or supported the positive outcomes mentioned above specifically. Questions like which element or feature in the AR game motivated students most during the learning process and why, which element or feature in the AR game helped students learn better during the learning process and why, which element or feature encouraged students to communicate to each other more and why, needed to be answered.

5.5 Suggestions for the Design of AR Learning Games

According to the reviewed studies, we can discern recommendations for the design of AR learning games that potentially lead to positive effects on students. During the design process, five aspects should be considered, which are learner groups, learning objectives, AR features, game mechanics, and social interactions.

CONCLUSION & FUTURE WORK

Conclusion

Robotoy is a game with real robots that you can control and watch through a mobile device or a personal computer. You can think about it as some kind of 'first person shooter' style of game with real world graphics. In the place of some 3d rendered scene you will see video streamed by a mounted camera on your real-world robot. Player controllers and robots communicates with each other through an existent LAN or through an Access Point setup in one of the robots. After the review of previous studies of AR learning games, we have interesting findings. most of the current AR learning games was played outdoors or in the classrooms. However, since students spend a lot of time at home and play digital games, it might be more effective to design AR learning games that can be played at home, which may encourage them to study spontaneously and in a more fun way. Finally, we found various game elements and AR features were used in the design of the AR learning games. However, there is a lack of systematic research on how different AR features and game elements influenced or supported the effects specifically.

Future Work

The following items should be developed sometime in future:

- Better looking pages in game.
- Provide movement sensors in each robot (such as gyros and accelerometers) and use it with some control algorithm (such as PID).
- Implement different alternative game goals (e.g. 'capture the flag').

In addition, we came up with five recommendations for the design of AR learning games to maximum the positive effects, which are: (1) involve learners in the design process, (2) always have clear learning objectives, and (3) design to encourage social interactions.

REFERENCES

1. Explosion animation were built with http://www.explosiongenerator.com/.
2. Animation sprites were built with http://css.spritegen.com/
3. Audio effects were downloaded from different authors at http://soundbible.com/
4. Laser sounds: http://soundbible.com/tags-laser.html http://soundbible.com/1771-Laser-Cannon.html
5. Explosion sounds: http://soundbible.com/107-Bomb-Explosion-1.html http://soundbible.com/1986-Bomb-Exploding.html
6. Damage sound: http://soundbible.com/1793-Flashbang.html
7. Sound API in JavaScript (Howler): https://github.com/goldfire/howler.js
8. Canvas Layers in JavaScript: https://bitbucket.org/ant512/canvaslayers/wiki/Home
9. Project idea: https://gustavohbf.github.io/robotoy/
10. Triebfurst, W.W.G.: ARVIKA: Augmented Reality for development, production and service. In: Proceedings of the 1st International Symposium on Mixed and Augmented Reality, IEEE Computer Society, Washington, DC, USA (2002)
11. Yuan, M.L., Ong, S.K., Nee, A.Y.C.: Assembly Guidance in Augmented Reality Environments Using a Virtual Interactive Tool. Singapore-MIT Alliance Symposium, Singapore (2005)
12. Green, S.A., Chase, J.G., XiaoQi, C., Billinghurst, M.: Evaluating the Augmented Reality Human-Robot Collaboration System. In: 15th International Conference on Mechatronics and Machine Vision in Practice M2VIP 2008, pp. 521–526 (2008)
13. 1. Akçayır, M., Akçayır, G.: Advantages and challenges associated with augmented reality for education: a systematic review of the literature. Educ. Res. Rev. 20,1 –11 (2017) 2.
14. Ayer, S.K., Messner, J.I., Anumba, C.J.: Augmented reality gaming in sustainable design education. J. Architectural Eng. 22(1), 04015012 (2016)
15. Azuma, R.: A survey of augmented reality. Presence Teleoperators Virtual Environ. 6(4), 355–385 (1997)
16. Bacca, J., Baldiris, S., Fabregat, R., Graf, S.: Augmented reality trends in education: a systematic review of research and applications. Educ. Technol. Soc. 17(4), 133–149 (2014)
17. Cascales, A., Laguna, I., Pérez López, D., Perona, P., Contero, M.: Augmented reality for preschoolers: an experience around natural sciences educational contents, pp. 103–112 (2012)

18. Freitas, R., Campos, P.: SMART: a SysteM of Augmented Reality for Teaching 2nd grade students. In: Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction, vol. 2, pp. 27–30, April 2008
19. Huizenga, J., Admiraal, W., Akkerman, S., Ten Dam, G.: Mobile game-based learning in secondary education: engagement, motivation and learning in a mobile city game: original article. J. Comput. Assist. Learn. 25(4), 332–344 (2009)
20. Ibáñez, M.B., Di Serio, Á., Villarán, D., Delgado Kloos, C.: Experimenting with electromagnetism using augmented reality: Impact on flow student experience and educational effectiveness. Comput. Educ. 71,1
21. Kamarainen, A.M., Metcalf, S., Grotzer, T., Browne, A., Mazzuca, D., Tutwiler, M.S.,Dede, C.: EcoMOBILE: Integrating augmented reality and probeware with environmental education field trips. Comput. Educ. 68, 545–556 (2013)
22. Schrier, K.: Using augmented reality games to teach 21st century skills. In: International Conference on Computer Graphics and Interactive Techniques, ACM SIGGRAPH 2006 Educators Program (2006)
23. Wu, H.K., Lee, S.W.Y., Chang, H.Y., Liang, J.C.: Current status, opportunities and challenges of augmented reality in education. Comput. Educ. 62, 41 –49 (2013)

APPENDICES

Appendices A

- To install noobs, visit www.raspberrypi.org/documentation/installation/noobs.md

Appendices B

- To install RoboToy program, visit github.com/gustavohbf/robotoy

GLOSSARY

HDMI:

"HDMI (High-Definition Multimedia Interface) is a proprietary audio/video interface for transmitting uncompressed video data and compressed or uncompressed digital audio data from an HDMI-compliant source device, such as a display controller, to a compatible computer monitor, video projector, digital television, or digital audio device. HDMI is a digital replacement for analog video standards." (Wikipedia HDMI)

Robot:

"A robot is a machine, especially one programmable by a computer, capable of carrying out a complex series of actions automatically." (Wikipedia Robot)

Sensor:

"A sensor is a device that detects and responds to some type of input from the physical environment." (Rouse)

SSH:

"Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network." (Wikipedia SSH)

VNC:

"Virtual Network Computing (VNC) is a graphical desktop sharing system that uses the Remote Frame Buffer protocol (RFB) to remotely control another computer." (Wikipedia VNC)