Mobile control interface for modular robots

Master Project, autumn 2011-2012
School of Computer Science

Jérémy Blatter
jeremy.blatter@epfl.ch

BioRob
Biorobotics Laboratory
Professor : Auke J. Ijspeert
Assistants : Julia Fink, Stéphane Bonardi & Rico Möckel

January 20, 2012
Abstract

This mobile control interface was developed to find an intuitive and efficient manner to interact with modular robots called Roombots. It is an iPad application that provides an augmented reality environment in order to interact with virtual 3D pieces of furniture made of Roombots. A user study was conducted to decide whether the ability to move inside the room and the use of augmented reality positively influence the interactions with the virtual objects. The users have to perform two tasks consisting in arranging furniture in a room. Their completion time and the precision of their placement are measured. It was found that the precision is improved when the users are able to move inside the room. The users said to prefer being dynamic and in an augmented reality environment, even if this last fact was not confirmed by the results of the study.

Acknowledgments

I would like to thank Professor Auke J. Ijspeert, who accepted me for this project, and all the Biorob members for their welcome. I would like to express my special gratitude to my supervisors Julia Fink, Stéphane Bonardi and Rico Möckel for their valuable advice and support, as well as Professor Pierre Dillenbourg and Patrick Jermann for their guidance. I would also like to thank all the people who has kindly accepted to participate to our user study.
Contents

Introduction 2

1 Background 3
  1.1 Roombots .......................................................... 3
  1.2 iPad 2 ............................................................... 5
    1.2.1 Hardware ...................................................... 5
    1.2.2 Application development .................................... 6
  1.3 Augmented reality applications .................................. 7
  1.4 Kinect ............................................................. 13
  1.5 OpenGL ES and OpenSceneGraph ................................. 13

2 iPad application 14
  2.1 Functionality ...................................................... 14
    2.1.1 Description ................................................... 14
    2.1.2 Interface and interactions ................................... 15
    2.1.3 Configuration screen ....................................... 16
    2.1.4 Logs .......................................................... 18
  2.2 Augmented Reality ................................................. 20
    2.2.1 Virtual reality vs augmented reality ........................ 20
    2.2.2 Device position ............................................. 21
    2.2.3 Device orientation .......................................... 23

3 User study 24
  3.1 Protocol .......................................................... 24
    3.1.1 Hypothesis and independent variables ..................... 24
    3.1.2 Test method ................................................ 25
    3.1.3 Dependent variables ...................................... 27
    3.1.4 Questionnaire .............................................. 28
  3.2 Results and analysis ............................................. 28
    3.2.1 Quantitative analysis .................................... 28
    3.2.2 Qualitative analysis ..................................... 31

4 Discussion 34
  4.1 Limitations ....................................................... 34
  4.2 Future perspectives .............................................. 35

Conclusion 37

Bibliography 38
Introduction

Roombots are modular robots developed at the Biorobotics Laboratory (BioRob) of EPFL. They are designed to be the basic units of adaptive pieces of furniture that can reconfigure themselves.

One of the goals of the Roombots project is to give the user the ability to easily arrange the furniture of a room thanks to the Roombots flexibility. Users need an intuitive and efficient interface in order to create or move a piece of furniture, and indirectly interact with the robots. The purpose of this project is to develop the required interface.

The main idea is to let the user arrange a virtual room with virtual pieces of furniture made of Roombots. In the future, the real robots will be able to construct the furniture in the real room as designed by the user.

As we wish to offer an immersive experience, we decided to develop a mobile application in which the user can place virtual models of furniture in an augmented reality environment. This provides the ability for the user to move inside the real room and to see it augmented by the virtual pieces of furniture with which the user interacts.

Among all the modern smartphones or tablet computers that could be used as augmented reality platforms, we chose to develop the application for the iPad 2, because of its excellent hardware components (display, sensors, camera, ...), its performance, the Mac OS X integrated development environment, and its commercial success.

In the first prototype of the iPad application developed in the scope of this project, the user can interact in a virtual or augmented reality environment with some predefined virtual pieces of furniture. The tracking of the iPad is done by its gyroscope for the orientation, and an external device, a Kinect, for the position. Then, in order to validate this application, we conducted a user study to assess how the ability to move and the use of augmented reality impact on the user experience and the efficiency of the application.

In the first chapter of this report, the background of the project is presented, both in terms of hardware and software.

The second chapter is dedicated to the iPad application. After an overview of its functionality, I focus on some key points related to the augmented reality.

Then, the third chapter examines the user study. The protocol is described in a first part, and the result and analysis in a second.

Finally, I discuss about the project in general, its limitations, and the future perspectives.
Chapter 1

Background

In this chapter, I review the hardware and software elements I used through this project, as well as the existing work related to it.

1.1 Roombots

Roombots (fig. 1.1) are modular robots, meaning that a Roombots structure is made of several robotic units. One module is composed of two spheres and has three degrees of freedom (fig. 1.2). Roombots modules can connect to each other using an active connection mechanism based on mechanical grippers and thus can change the shape of the structure over time, by reconfiguring themselves.

\hspace{5cm}

Figure 1.1: Three Roombots modules\textsuperscript{1}

\hspace{5cm}

Figure 1.2: Degrees of freedom\textsuperscript{1}

\textsuperscript{1}rendered picture created by A. Spröwitz
A Roombots structure can integrate some passive parts to form bigger shapes such as pieces of furniture (fig. 1.3). As the structure is able to self reconfigure, a table could for example become a chair.

![Roombots furniture](image1.png)

Figure 1.3: Roombots furniture

There are three main research directions in the Roombots project: the reconfiguration, the locomotion, and the user interface, which is related to this project.

So far, there exists an interface to view and design the furniture models using cubes as basic units (fig. 1.4a), and some to control individually each degree of freedom of the robots (fig. 1.4b). But these interfaces are made for experts only, that is the reason why we wish to provide in this project a new type of interface for the end users.

![Interfaces for Roombots](image2.png)

Figure 1.4: Interfaces for Roombots

---

\(^2\)created by A.Spröwitz
\(^3\)created by S.Bonardi
1.2 iPad 2

I present the hardware configuration of the iPad, and the way to develop an application for it.

1.2.1 Hardware

The iPad is a tablet computer developed by Apple since 2010. The iPad is running the operating system iOS. The latest version of the iPad, which is used in this project, is the iPad 2, released in March 2011. It has been designed as a multimedia platform (books, movies, music, games, web, ...) but some of its characteristics are appropriate for an augmented reality application:

Display: The iPad 2 has a 25cm (9.7") multitouch display at a resolution of 1024 by 768 pixels (4:3 aspect ratio).

Camera: The camera is a 720p HD camera (resolution of 1280 by 720 pixels, 16:9 aspect ratio) with a frame rate around 30 fps. When holding the device in landscape orientation, the horizontal field of view is 44.5 degrees.

There are two ways to use the camera in an application. We can either display the real time video directly in a view of the application, or access to an array of pixels for each frame (more details in section 2.2.1).

3-axes gyroscope (example of the iPhone on fig. 1.5a): The gyroscope gives various information at a maximum rate around 100 Hz. It is possible to get the raw rates of rotation around the three axes, or the mathematical representation of the attitude of the device as Euler angles (roll, pitch and yaw), a rotation matrix, or a quaternion$^4$.

A timestamp representative of the time at which the gyroscope event occurred is also provided.

3-axes accelerometer (example of the iPhone on fig. 1.5b): The accelerometer gives the value of the acceleration in [g], which is the gravity on earth ($1[g] = 9.81[m/s^2]$), along the 3 axes of the device at a maximum rate around 100 Hz and in a range of [$-2g$, $+2g$].

A timestamp representative of the time at which the accelerometer event occurred is also provided.

Wi-Fi connection: The iPad 2 has a Wi-Fi card supporting the 802.11a/b/g/n standard. Thus it can communicate with a computer or another device via BSD sockets.

Other: The iPad 2 has also a GPS and a magnetometer, but I didn't use them.

$^4$Quaternions are a way to describe rotation transformations in computer graphics. Please refer to http://www. openg esegraph.org/ projects/ osg/ wiki/ Support/ Maths/ QuaternionMaths for more details about quaternion in that scope.
1.2. IPAD 2

CHAPTER 1. BACKGROUND

(a) gyroscope

(b) accelerometer

Figure 1.5: iPhone sensors axes in landscape orientation

1.2.2 Application development

To develop an application for the iPad, we have to use a Mac computer and its integrated development environment Xcode. The version of Xcode used at the beginning of this project was Xcode 4.1 with iOS SDK 4.3 corresponding to iOS versions 4.3 to 4.3.5. Xcode 4.2 was then released on October 12, 2011 concurrently with the release of iOS 5.0 and iOS SDK 5.0. Finally, the actual version iOS 5.0.1 was released on November 10, 2011.

The programming language is Objective-C 2.0, a revision of Objective-C, which is an object-oriented programming language based on the C89 language.

I have to include several frameworks of the iOS SDK in order to have access to the specific parts of hardware I need. For example, AVFoundation is necessary for the camera, CoreMotion for the sensors, CFNetwork for the Wi-Fi, OpenGLES, CoreMedia and CoreVideo for displaying OpenGL objects on the screen, ...

\[\text{adapted from http://developer.apple.com/library/ios/#documentation/EventHandling/Conceptual/EventHandlingiPhoneOS/MotionEvents/MotionEvents.html}\]
1.3 Augmented reality applications

The purpose of the augmented reality is to provide to the user a view of the real world augmented with some virtual objects, labels, videos, ... and most often in real time. This concept is in contrast to the one of virtual reality, where the real world is replaced by a fully virtual world.

I give below an overview of iPhone/iPad applications dealing with augmented reality. I would like to describe their main functionality and how they use the sensors of the device.

One can classify them in two categories regarding if they use markers or only the sensors and the GPS of the device.

Applications based on markers

Markers and image processing are used to compute how to display the virtual object at the correct position and with the correct orientation:

- **Arusma**\(^6\) (fig. 1.6) and **Zappar**\(^7\) (fig. 1.7) allow to display a video or a 3D animation instead of a static image which is used as the marker. For example, it can display the teaser of a movie on its poster, an advertisement on the logo of a company or a video report on a newspaper article.

- **AR-Robot**\(^8\) (fig. 1.8) and **String**\(^9\) (fig. 1.9) display a 3D animated object (robot, dragon, shoe, ...) on a picture/marker. In String, we can even interact with the objects, by drawing 3D lines or making a little monster walk outside of the picture. These are probably the most robust applications among those I have tried, either for marker detection, movement smoothness, maximum angle and distance, ...

---

\(^6\)http://itunes.apple.com/gb/app/aurasma-lite/id432526396
\(^7\)http://itunes.apple.com/us/app/zappar/id429885268
\(^8\)http://itunes.apple.com/fr/app/ar-robot/id430608785
\(^9\)http://itunes.apple.com/fr/app/string-augmented-reality-showcase/id417606596
Applications based on sensors and GPS

These applications don’t use any markers, but only the device sensors:

- **Wikitude**\(^{10}\) (fig. 1.10) or **Panoramascope**\(^{11}\) (fig. 1.11) can identify some places near the user. Wikitude displays a lot of information, such as famous buildings, restaurants, Wikipedia articles, photos, events, ... Panoramascope draws the skyline and displays the name of the mountain’s peaks.

Both uses the GPS, the magnetometer and the gyroscope (if available, otherwise the accelerometer) to get the position and the orientation of the device and to display the labels in the right position on the real time camera view. Panoramascope makes some image processing on the camera view to superimpose the skylines.

---

\(^{10}\)http://itunes.apple.com/us/app/wikitude-augmented-reality/id329731243

\(^{11}\)http://itunes.apple.com/us/app/panoramascope-augmented-reality/id341446556
1.3. AUGMENTED REALITY APPLICATIONS CHAPTER 1. BACKGROUND

- **Spyglass**[^12] (fig. 1.12) or **Theodolite HD**[^13] (fig. 1.13) are navigation toolkits with many features, especially a gyro compass, gyro horizon and location tracker in augmented reality. They use every sensor of the iPad 2 (GPS, magnetometer, gyroscope, accelerometer and camera) to reach a high level of accuracy.

![Spyglass](image1)

![Theodolite HD](image2)

Figure 1.12: Spyglass[^12] (iPhone version)  Figure 1.13: Theodolite HD[^13]

- **Sky-Siege**[^14] (fig. 1.14) allows the user to control a virtual helicopter in augmented reality from a fixed point. The gyroscope is used to get the orientation of the device, whereas the position is not needed.

![Sky-Siege](image3)

Figure 1.14: Sky Siege[^14]

1.3. AUGMENTED REALITY APPLICATIONS

CHAPTER 1. BACKGROUND

Furniture applications  I detail three marker free applications that are in the field of furniture:

- **Atelier Pflister**\(^{15}\) (fig. 1.15) is an application that allows the user to place virtual pieces of furniture in a real room. It provides a great user experience, but we can only place the furniture into a static picture of the real room, and not with the real time camera view. However, this allows to use a view from above or a plan of the room and to place the furniture in 2D as we could do in a classic interior design software like Sweet Home 3D\(^ {16}\).
  The interactions, that are translations (drag with one finger), rotations (rotate two fingers) and scale of the whole scene (pinch gesture) work well, even if the translation may not be very handy because the object is following the finger. Also, as there is no indication about which object is selected, it is sometimes unclear which object we are interacting with. While moving an object, there is no collision detection with the real room or other virtual objects.
  The scene seems very realistic because the pieces of furniture are well detailed, have a simple shadow, and their orientation is correctly estimated. This estimation is probably done by computing the gravity vector in the iPad referential, using the accelerometer data at the time the picture is taken.

- **uDecore**\(^ {17}\) (fig. 1.16) provides the placement of the furniture with the real time camera view, but, as only the pitch rotation of the device is managed (with a quite important delay...), the objects are always moving in the real room and their orientations are not perfect, so the user experience is not really satisfying. Mainly, the yaw rotation and an estimation of the height of the device or the ability to scale are missing. The reason is probably that this application is using only the accelerometer to compute the gravity vector, and that is not sufficient to get the yaw or the height as this is only a vertical vector.
  There is no collision detection while moving objects and no shadows below them, so the furniture gives the impression of floating.
  The translations (drag with one finger) and rotations (rotate two fingers) works well (even if the objects tremble a little and sometimes disappear) and the furniture are quite detailed, but the overall scene does not seem very realistic due to the problems mentioned before.

- **Snapshop**\(^ {18}\) (fig. 1.17) provides a real time camera view too, but the user experience is very bad. First, we can place only one big piece of furniture which is not a 3D model, but a 2D picture. Then, none of the device’s movement is supported, so the object is always moving in the real room and its orientation is bad (nearly a front view). The rotation (rotate two fingers) does not work well and is useless because it rotates the picture on a vertical plane. The scale (pinch gesture) is a strange mix between object scaling and camera zoom... And there is no collision detection neither shadows.

\(^{15}\)http://itunes.apple.com/us/app/atelier-pflister/id385703066
\(^{16}\)http://www.sweethome3d.com
\(^{17}\)http://itunes.apple.com/gb/app/udecore/id449926730
1.3. AUGMENTED REALITY APPLICATIONS

CHAPTER 1. BACKGROUND

Figure 1.15: Atelier Pfister

Figure 1.16: uDecore

Figure 1.17: SnapShop

Synthesis

From this short review, I can define three main requirements for a good augmented reality application:

1. Display virtual objects or information above the real time camera view of the device
2. Track the position of the iPad. This position might be more or less precise (from centimeters to meters) depending on the goal of the application. Sometimes, tracking the position is even not necessary if the application imposes enough constraints
3. Track the orientation of the iPad. This is the attitude of the device in space, and can be seen as the 3D direction vector at which the user is looking through the device camera

I explain how I satisfied these three requirements for my application in the section 2.2.
Moreover, I can define some guidelines for my application. First of all, I should develop a marker free application, so it could be used in a more natural environment (i.e. in any room) without constraints on the placement of the furniture. Then, I should use the real-time camera view as background, and the sensors to compute the position and the orientation of the device. The position needs to be accurate, maybe of the order of a few centimeters, and the orientation should take into account at least the pitch and the yaw rotation without delays.

As interactions, the translation and the rotation are necessary, whereas the scale might sometimes be useful. The translation with which the furniture is not stuck to the finger seems more convenient. I should also give a visual feedback to the user, so he knows which object he is interacting with.

A collision detection system does not seem to be required in order to have a realistic application. On the contrary, providing well detailed pieces of furniture with a shadow and a good orientation is important.

**Frameworks**

To conclude this section, I present some useful frameworks for iOS device:

**Augmented reality** There exist many augmented reality frameworks for iOS, but they all use markers to position a virtual object in the real scene. As I don’t wish to make a augmented reality application based on markers, I did not use any of these frameworks. However, some of them are listed below:

- Qualcomm’s augmented reality SDK\(^{19}\)
- Metaio Mobile SDK\(^{20}\). Note that this framework allow 3D tracking without marker in its pro version.
- ARToolWorks : ARToolKit for iOS\(^{21}\)
- NyARToolkit\(^{22}\) that can be compiled and use with iOS\(^{23}\)
- SGAREnvironment\(^{24}\)

**Image processing** A well known library for image processing is OpenCV and it is available as a framework for iOS\(^{25}\). But I finally did not make any image processing in the application, so I do not need this framework any more.

---

\(^{19}\)https://developer.qualcomm.com/developer/mobile-technologies/augmented-reality

\(^{20}\)http://www.metaio.com/software/mobile-sdk/

\(^{21}\)http://www.artoolworks.com/products/mobile/artoolkit-for-ios/

\(^{22}\)http://nyaita.jp/nyartoolkit/wp/?page_id=188


\(^{24}\)https://github.com/simplegeo/SGAREnvironment

1.4 Kinect

The Kinect (fig. 1.18) is a device developed by Microsoft for the Xbox 360 video game console since 2010. It uses a camera and an infrared sensor to detect the motion and the depth of the user, in order for them to control the game thanks to their body gestures and without the need of a game controller. Microsoft has released a SDK in June 2011, so it is possible to develop our own applications using the Kinect.

In my application, I need the position of the user given by the Kinect to estimate the position of the iPad, as detailed in section 2.2.2.

![Figure 1.18: Kinect](image)

1.5 OpenGL ES and OpenSceneGraph

OpenGL\(^{27}\) (Open Graphics Library) is a cross-platform and cross-language API managed by the Khronos Group. It is widely used to produce 3D graphics, particularly in video games.

The iPad 2 and its operating system iOS support only the version OpenGL ES 2.0\(^{[2]}\), which is a subset of OpenGL especially designed for embedded systems.

OpenSceneGraph[4] (OSG) is an open source 3D graphics toolkit written in C++ and based on several OpenGL versions. Since the OSG version 3.0 released in June 2011, OpenGL ES 2.0 is supported, therefore it is possible to use OSG with an iOS device.

The sources of OSG for iOS are available on Github\(^{28}\). As said in the readme file, the sources must be built with CMake to generate static libraries which will be included in the Xcode project.

As the programming language for the iPad is Objective-C and OSG is written in C++, I have to write some of the files in Objective-C+++, which is a combination of Objective-C and C++ with .mm files extension.

I use OSG and some of its features (object importation, multitouch event recognition, ...) to construct and interact with the 3D scene within the application.

\(^{26}\)http://www.xbox.com/fr-FR/Xbox360/Accessories/kinect/Home

\(^{27}\)http://www.khronos.org/openg1

\(^{28}\)https://github.com/stmh/osg/tree/iphone

![Figure 1.19: OpenGL ES](image)

![Figure 1.20: OpenSceneGraph](image)
Chapter 2

iPad application

This chapter describes first the main functionality of my application, and then explain how to make augmented reality on such a device.

2.1 Functionality

The iPad application developed in the scope of this project is called “Roombots AR” and is available only for the second generation of iPad, the iPad 2, because the back camera and the gyroscope are necessary. After a brief description of the application, I detail its interface and the provided interactions.

2.1.1 Description

Roombots AR allows the user to interact with a collection of given virtual pieces of furniture, either in an augmented reality (AR) or a virtual reality (VR) environment.

For this prototype of the application, the models of the furniture are made of cubes, and not of Roombots, mainly for performance reasons explained in the section 4.1 about the limitations.

The 2 following screenshots give an example of a scene created by the application in both augmented reality (fig.2.2a) and virtual reality (fig.2.2b).

Figure 2.1: Roombots AR icon

Figure 2.2: Screenshots of a scene with two chairs
2.1. FUNCTIONALITY

When the user moves inside the room, the iPad has to compute in real time how to display the 3D scene in order to respect the new position and orientation of the user, and to give him the impression that the furniture are fixed inside the room. For that purpose, it uses its sensors and the data given by an external device, a Kinect. More detailed explanations about how the user movements are managed can be found in sections 2.2.3 and 2.2.2.

2.1.2 Interface and interactions

The interface of the application (fig. 2.3) can be divided into three main parts:

- the room in the background, either the real time camera view or a 3D representation of it
- the two HUDs\(^1\) placed on the top and at the bottom of the screen
- the 3D scene containing the furniture added by the user, and possibly some existing pieces of furniture

The available interactions with the furniture that are provided by this interface are:

**In the bottom HUD**

**creation** : Single tap on the button corresponding to the desired type of furniture (chair or table). It will appear one meter in front of the user, except if a collision is detected during the creation, in which case the piece of furniture is placed at the position of the user.

**deletion** : Single tap on the button with a red cross to delete the selected piece of furniture.

\(^1\)Head Up Displays : fixed parts of the user interface used to display either passive (labels) or active (buttons) pieces of information to the user.
In the top HUD

**color change**: Single tap on the desired color to change the color of the selected piece of furniture. The furniture are white by default.

**Wi-Fi label**: This is only an indicative label that appears when the application needs to receive data by Wi-Fi. It is green as long as the reception of the packets is correct, and turns red in case of problem.

In the main window

**selection**: Single tap on a piece of furniture to select it. A red arrow appears on top of the selected object, as well as the HUD buttons corresponding to actions that require a selected piece of furniture: deletion and color change.

**deselection**: Single tap on any space without object. The red arrow and the HUD buttons corresponding to deletion and change color disappear.

**translation**: Drag with two fingers to translate the selected object in 2D on the floor. The object is not stuck to the finger.

Collisions with the walls and other existing objects in the room are handled, both in AR and VR, if the coordinates of the room has been specified beforehand. The collision detection is very simple, as each piece of furniture is view as a sphere. In case of collision, the translation of the selected object is stopped in the direction of the colliding object, so they cannot overlap, but the selected object can “slide” along the other.

However, notice that the collisions between the pieces of furniture added by the user have been disabled because when the scene becomes crowded, it is difficult and frustrating to place the objects.

**rotation**: Drag horizontally with one finger to rotate the selected object around a vertical axis.

2.1.3 Configuration screen

This screen (fig. 2.4) has been added for the user study and should not be seen by the final users. It allows my to check the success of the set up of the application and the various hardware elements, as well as setting some options for the experiment.

In a final version of the application, this screen should be removed. The application should produce error messages only in case of failure during the set up, and the options should be available from a option menu or panel inside the application.
2.1. FUNCTIONALITY

CHAPTER 2. IPAD APPLICATION

The options are:

- **User ID**: an identifier for the user. This identifier is indicated in the log files names (see section 2.1.4).

- **Condition** (see the user study test method at 3.1.2):
  - **Augmented Reality**: if yes, initialize the camera for loading the augmented reality environment, otherwise load the virtual reality environment (see section 2.2.1 for more details).
  - **Ability to Move**: allow or not the user to move inside the room, that is to change his position (see section 2.2.2 for more details). If yes, initialize the Wi-Fi connection. Notice that the orientation of the user is managed in both case.

- **Start position (X, Y, Z)**: the initial position of the OSG camera. It is necessary if the ability to move is disabled.

- **Task Number**: choose between the two proposed tasks (see the user study test method at 3.1.2).

There is also a check on the initialization of the gyroscope. As this sensor is used in any case to manage the orientation of the user (see section 2.2.3 for more details), it is always started with the application and cannot be stopped, otherwise the application won’t work well.
2.1.4 Logs

During a run of the application, every action of the user is recorded and written into a file, so it is possible to study his behavior afterward, as I describe in the analysis of the user study at section 3.2.

The files have a .plist extension (means Property List). This is the file type used in Mac OS and iOS to store serialized objects. They use the XML format, so they can be read like any .xml file.

The easiest way to write these files is to add into an array a String object describing the action and a timestamp relative to the beginning of the run. Then, when the run is stopped, these arrays are written into files, which are placed in the “document” folder of the iPad and can be transferred on a computer via Xcode or iTunes.

The logs are divided into two files. The movements of the user (i.e. its position and orientation) are written in the camera file named “cam.plist”, whereas the actions of the user are in the user action file, named “us.plist”. The full name and path of these files indicate some information about the run, such as the identifier of the user, the start date and time and the condition chosen. The format for the two files names is the following:

1. [user ID]_[year]-[month]-[day]_[hour]-[minutes]-[seconds]_[condition]#[task]_cam.plist
2. [user ID]_[year]-[month]-[day]_[hour]-[minutes]-[seconds]_[condition]#[task]_us.plist

For example, we may have these two files for a run in condition SA and Task #2:

1. 8_2011-12-15_10-42-55_SA2_cam.plist
2. 8_2011-12-15_10-42-55_SA2_us.plist

In the camera file, a line is added at each frame (about 30 per second) and indicate the position of the OSG camera and the attitude of the iPad with the format:

<table>
<thead>
<tr>
<th>time[s]</th>
<th>pos_x(m)</th>
<th>pos_y(m)</th>
<th>pos_z(m)</th>
<th>att_roll[rad]</th>
<th>att_pitch[rad]</th>
<th>att_yaw[rad]</th>
</tr>
</thead>
</table>

The relation between the position of the OSG camera, the attitude of the device, and the position and orientation of the user is explained at sections 2.2.3 and 2.2.2.

The format used for the action file depends on the type of action. For all of them, the identifier of the piece of furniture (furn_id) is given after the time and a three letter code:

1. furniture creation : time[s] CRE furn_id type pos_x[m] pos_y[m] pos_z[m] angle[rad]
2. furniture deletion : time[s] DEL furn_id
3. furniture translation : time[s] MOV furn_id pos_x[m] pos_y[m] pos_z[m]
4. furniture rotation : time[s] ROT furn_id angle[rad]
5. change furniture color : time[s] CC0 furn_id style
6. furniture selection : time[s] SEL furn_id
7. furniture deselection : time[s] DSL furn_id
Note that the rotation angle is given in the room referential, and in our setup for the user study, an angle of 0 radian means that the piece of furniture is heading at the -Y axis.

At the end of the user actions file, there is a summary of the final furniture state after a blank line. That is, for each piece of furniture added by the user and still present at that moment, there are these 3 lines:

1. `time[s] TYP furn_id code`
2. `time[s] MOV furn_id pos_x[m] pos_y[m] pos_z[m]`
3. `time[s] ROT furn_id angle[rad]`

The code is the concatenation of the type and the color of the furniture, so it describes uniquely the kind of furniture. For example, a code of 11 means a white (1) chair (1), and a code of 24 means an orange (4) table (2).

A possible short user action file can be found as example in the appendices.

We also wanted to record the screen of the iPad in order to have a live (or at least differenced) view of what the user is doing, but there is no easy solution. We thought of adding a camera on the user’s shoulder, but it should have been too disturbing for him. Another idea we had, was to regularly send by Wi-Fi a screenshot of the application, but this was not feasible due to time constraints, and would have possibly caused technical issues. Indeed, that should have slow down the application and require too much Wi-Fi bandwidth. We could also save the screenshots locally and have access to them afterward, but we finally though that the logs and a external fixed camera would be enough.
2.2 Augmented Reality

In this section, I explain the strategy used for making augmented reality by fulfilling the three requirements I defined in section 1.3:

1. Display the virtual scene above the real time camera view
2. Track the position of the device
3. Track the orientation of the device

The last two points are crucial. As the user is able to move freely inside the room with the iPad, all its movements should be reported into the application to keep the match between the real room and the virtual scene. Indeed, every change of the position or orientation of the device, caused by a movement of the user, change the point of view of the real room. Thus, the point of view of the virtual scene should change the same into OSG.

The figure below will be the basis of my explanations. It shows the steps between the various sensors (blue rhombus) to the layers displayed on the iPad screen (yellow parallelograms) as they are in the last version of the application.

2.2.1 Virtual reality vs augmented reality

I said in the description that the application could be used either in an augmented reality or a virtual reality environment (screenshots on fig.2.2a and 2.2b). I explain now the difference between these two concepts from a technical point of view.

Virtual reality

In virtual reality, a fully virtual world replace the real world. Thus, I have to construct a 3D model of the room and the existing furniture inside it. I use OSG to draw and render this model.
2.2. AUGMENTED REALITY

This process is equivalent to the construction of a 3D environment for any 3D application or video game, more detailed explanations are beyond the scope of this report. Please refer to OSG documentation [4] or book [6], or any OpenGL tutorial for more details.

Augmented reality

In augmented reality, I have to superimpose some 3D objects to the real time camera view. There are mainly two ways of doing that:

1. The camera view is managed by OSG
   The data of the iPad camera, that is an array containing the value of all the pixels for each frame, is delivered in real time by the system at a maximum frame rate of 30 fps. Then, the application transfers these data to OSG, which constructs a texture from this array and bind it on a 2D shape placed as the background of the scene.
   This is the easiest way, chosen at the beginning, and still present in the version of the application used during the user study. The main drawback of this technique is that if the 3D scene is longer to render than 1/30s, some camera frames are dropped and the display might look very jerky.

2. The camera view is managed by Objective-C
   It is possible to get directly a Objective-C View object containing the real time preview of the iPad camera, at a maximum frame rate of 30 fps. This view is added as a subview of the window as well as the OSG scene that is rendered in another subview of the window, placed on top.
   I have to be careful to set the background of the OSG scene transparent in order to see the iPad camera preview behind. As this is not yet provided by the last version of OSG, it is necessary to modify manually the sources and compile them again. Explanations are given by Stephan Maximilian Huber, who did the OSG port for iOS, in a thread that I created on the OSG forum.
   As the real time camera view is no longer managed by OSG, the drawback of the first technique is by-passed and the camera view is always displayed at about 30 fps. This technique is better, so it is the one represented on the diagram above, and the last version of the Roombot AR application works this way.

2.2.2 Device position

I decided not to use any markers to compute the position of the iPad, but only its sensors. I needed several trials to obtain an acceptable result:

iPad accelerometer

My first intuition was to use only the data from the accelerometer. The iPad 2 with iOS 4 provides directly the user acceleration by removing the gravity from the raw acceleration measured by the sensor. Then, basic physics says that we can find the real time position via a double integration, given an initial position and speed. Indeed, the derivative of the position is the speed, which is the derivative of the acceleration.

I expected that this result would be accurate enough to be exploited right away, even if another technique (image processing for example) may be necessary to adjust them. But unfortunately, this is not the case. I obtain very noisy and badly calibrated data, so in the application, the scene moves all the time without control...

---

2 http://forum.openscenegraph.org/viewtopic.php?t=9472
I tried to calibrate the accelerometer within the application at startup, or to extract manually the gravity from the raw data of the accelerometer thanks to the attitude of the device given by the gyroscope. The results are better, but still unusable...

It seems that to compute precisely the position of the device only with its sensors, it is necessary to make a sensor fusion, as some of the application presented in section 1.3 do. The principle of the sensor fusion is to get the data from all the sensors of the device (accelerometer, magnetometer, gyroscope, camera, GPS, ...) and combine them together in order to have the most accurate value possible. But this is a difficult and quite new technique for interactive devices such as the iPad, so it cannot fit into the scope of this project. More details about sensor fusion in robotics can be found in the Springer Handbook of Robotics[5], chapter 25.

In conclusion, I had to find another solution.

Markers and image processing
Among all the possible solutions we discussed at that moment, including the use of an external tracking system such as a Kinect, cameras or Wi-Fi terminals, we decided to use markers and image processing with OpenCV.

I quickly managed to get marker detection working on the iPad, but at the cost of a huge drop in frame rate, even if I set the camera to a medium resolution (960 by 540 pixels). Moreover, some tests we did to compute the position of the device from three markers placed on the ground was not convincing in terms of precision. Despite many attempts to calibrate the camera of the iPad, we had at best an error of the order of a meter.

Also, it was difficult to define the best strategy for the placement of the markers all over the room, in order to compute a unique 3D position. And this large number of markers would have made the room very unnatural.

For all these reasons, we finally give up the idea of markers...

Kinect
We tried then to compute a good estimation of the position of the device with a Kinect. This is only an estimation, because the only thing that we can track out of the box with an external device such as a Kinect is the body of the user, and not an object like the iPad. Moreover, as we have only one Kinect, the iPad is not always visible by the Kinect according to the orientation of the user.

To compute this estimation, we used an existing code from the Nestk library³, which tracks the skeleton of the user, and provides the 3D positions of some joints. Stéphane Bonardi has adapted this code to compute the position of the iPad as the position of the user’s neck, plus 30cm in the direction of the normal vector of the plane containing the torso and the two shoulders of the user.

Finally, these values are sent in real time by Wi-Fi to the iPad via a BSD socket with the UDP connectionless protocol. All these steps can be seen on the diagram of the figure 2.5.

This estimation we got thanks to the Kinect is acceptable for our needs, of the order of 10 centimeters, but still has some drawbacks :

- The user must stay in the field of view of the Kinect at any time, so we had to restrict his movements inside the room to a predefined area of about 3 x 2.50 meters
- As the Kinect tracks the user and not the iPad, it cannot be detected by the application if the user moves the iPad without moving himself, for example by extending his arms

³http://nicolas.burrus.name/index.php/Research/KinectUseNestk
• The signal is nevertheless a bit noisy, so the virtual scene trembles sometimes. Moreover, the height might be varying depending on the distance between the user and the Kinect, because this is difficult to get a perfect Kinect calibration. Indeed, the depth measurement given by the Kinect is not linear with the distance\(^4\). And as the users do not all have the same height, we did our best to adapt the tilt of the Kinect (front or back) for each user in order to minimize the error. We noticed a maximal error around 5cm.

### 2.2.3 Device orientation

When I have detailed the hardware of the iPad at section 1.2, I said that we can get the attitude of the device as a quaternion. This quaternion can directly be used in OSG as a transformation, so no other computation is required. This provides the three axes of rotation: roll, pitch and yaw.

With the first tests, that technique seems to work very well, but we noticed during the pretests of the user study that a drift can occur during a long run. As this drift depends mainly on the rotation rate of the device, it is unpredictable and can not be easily corrected. However, due to time constraints, we had to keep it unchanged for the user study, during which we were especially attentive to a possible gyroscope issue.

\(^4\)http://mathnathan.com/2011/02/03/depthvsdistance/
Chapter 3

User study

In this chapter, I present the user study we conducted for the Roombots application, and the results we obtained. In the first section, I go through the steps of the preparation of the study, from the hypothesis to the sequence of the experiment. Then, I describe in a second part the results and their analysis.

The study has been done with the help of my supervisors: Julia Fink, Stéphane Bonardi and Rico Möckel.

3.1 Protocol

3.1.1 Hypothesis and independent variables

The purpose of this study is to openly explore how users interacted with the device and the application. More concretely, we want to see what are the difficulties they have to interact with virtual 3D pieces of furniture on iPad, and decide what could improve these interactions. We defined two research questions:

1. Does the ability for the user to move inside the room in order to place the pieces of furniture influence the result of the experiment?
2. Does the use of an augmented reality environment as opposed to virtual reality (a pure 3D representation) influence the result of the experiment?

These two questions are the basis of the modalities we wish to oppose. The modalities are called independent variables and are described as:

**Ability to move**

1. static: the iPad is fixed on a stand and the user can only rotate it horizontally and vertically (yaw and pitch)
2. dynamic: the user can move freely with the iPad in an area defined by a tape on the floor (as seen on figure 3.1)

**Room representation**

1. augmented reality: the user will perform in an augmented reality environment
2. virtual reality: the user will perform in a 3D virtual reality environment
In order to cover all the modalities, we must define four conditions as shown in the table 3.1. The control condition is static and virtual.

<table>
<thead>
<tr>
<th></th>
<th>Virtual</th>
<th>Augmented</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>SV (Control)</td>
<td>SA</td>
</tr>
<tr>
<td>Dynamic</td>
<td>DV</td>
<td>DA</td>
</tr>
</tbody>
</table>

Table 3.1: Conditions

3.1.2 Test method

We recruited 24 subjects (eight women and sixteen men), who were all students, researchers or employees at EPFL, from 19 to 50 years old (mean age 25), and various background (biology, computer science, physics, chemistry, ...).

We made a within subject experiment, meaning that each subject was assigned to two conditions, with one independent variable fixed. For example, a subject can test both static and dynamic in augmented reality (group SA/DA), or both virtual and augmented reality in dynamic (group DV/DA). This way, the two conditions differ by only one modality, which is easier for the subject to compare the two conditions he had tested, and for us to analyze the results. Thus, we split the subjects into four groups of six subjects (two women and four men). The groups are called SV/SA, SA/DA, DV/DA and SV/DV.

To avoid the learning effect, half of each group had to start with the second condition, then the first.

![Figure 3.1: Set up of the experiment](image)

The experiment consist of two small tasks in each of the two conditions. The sequence is:

1. The subject was welcomed and given a presentation of the Roombots project in order to have a background of what we try to do and get familiar with the environment.

2. The subject was asked to has to sign a consent form (given in the appendices), mainly to accept to be video taped and audio recorded during the experiment.
3. The subject received a short introduction to how the application is working. In the static virtual condition, I explained how to create, move, rotate, change the color, select, deselect and delete a piece of furniture. Then, I gave the subject one minute to try and feel comfortable with it.

4. For each condition, we gave some instructions to the subject, such as how to rotate the stand in static, or where he can move in dynamic.

5. I explained each task, with the help of a scheme on the whiteboard (fig. 3.2 and 3.3):

**task #1**: construct a circle of ten chairs, two of each color, with a table in the middle. The order of color for the chairs as well as the color of the table are free of choice, and the pieces of furniture could be placed anywhere in the room.

**task #2**: place a chair on each side of the three virtual objects present in the room, alternating the colors between red and blue and orientating the chairs towards the middle of the room.

![Figure 3.2: Scheme of the task #1](image)

![Figure 3.3: Scheme of the task #2](image)

The Task #1 shows if the subject is able to organize in a logical manner several virtual objects in the room, whereas the task #2 tests the ability of the subject to mentally reconstruct the environment of the room to deal with existing objects.

6. I give the iPad to the subject. The application is already started and configured for the task. During the completion of the tasks, we asked subjects to give us their thoughts and feelings.

7. At the end, the subject has to fill in a questionnaire during a short interview (more details in section 3.1.4).

8. The subject is thanked for participating and received two coupons for campus food as a reward.
3.1.3 Dependent variables

With the logs recorded by the application (see section 2.1.4 for technical details), we can measure some variables for each condition:

- The displacement of the subject inside the room
- The completion time for each task, measured between the first and the last subject action.
- The precision of the final placement:
  - for the task #1, we construct a normalized index by summing the normalized values of two measured variables (fig. 3.4):
    - the standard deviation of the distance between the chairs and their centroid, to indicate the roundness of the circle of chairs
    - the distance between the centroid of the chairs and the center of the table, to indicate the symmetry of the whole scene
  - for task #2, we measure the distance between the position of the chairs and their exact theoretical position (fig. 3.5)
- The distortion of the ellipse that fit best the positions of the chairs in task #1. This is another indicator of the roundness of the circle of chairs, but it looks more at the general shape and not the shift of every single chair
- The number of errors for each task. An error can be a missing piece of furniture or one in excess, a wrong furniture type or color, a collision between two objects, or the number of delete.
- The number of actions per type for each task. We count the number of creations, deletions, selections, rotations and movements.

Figure 3.4: Possible placement in task #1.
The black line represent the tape. The pink star is the position of the stand. The big green square is the table and the pink circle is its center. The small blue squares are the chairs. The black cross is the centroid of the chairs and the black circle is centered on the centroid with a radius equals to the mean distance between the chairs and their centroid

Figure 3.5: Possible placement in task #2.
The gray area is the area delimited by the tape. The pink star is the position of the stand. The black squares are the existing virtual objects and the colored squares are the chairs
3.2 Results and analysis

The purpose of this study is to explore how users interacted with the system and to define if the ability to move and the use of augmented reality can improve the interactions with 3D furniture in an iPad application like Roombots AR. To that end, we defined four modalities that we wish to oppose, some variables that we measured during the experiment, and a questionnaire.

To group all the collected data and easily have access to those we would like to analyze, we have constructed a SQLite\textsuperscript{1} database that we can access from R\textsuperscript{2}, one of the most used software for statistical computing.

To oppose the modalities, we run ANOVA tests, which means analysis of variance. This is a statistical test to verify whether a variance is statistically significant, thus did not occur by chance. The result of the test is called the p-value and indicate the probability that the null hypothesis, saying that all the samples follow the same normal distribution, is true. If the p-value is lower than a given level of risk alpha (here, we define alpha as 0.05), we can reject the null hypothesis and conclude that the samples are significantly different with the level of risk alpha.

To use an ANOVA test, the sample must verify some assumptions, including the normality of the distribution. By examining our samples, this is not exactly the case, but we think that mainly comes from the small size of our samples.

3.2.1 Quantitative analysis

For the quantitative analysis, we focus on the completion time and the precision of the final placement, because the other variables we measured are not usable. Indeed, there were too few errors to make statistics on it, and the number of actions cannot be analyzed easily. Indeed, it is not sufficient to look at the final number of actions of each type, but we should also analyze when and which object is concerned. Thus, we could capture some interesting information, such as the trial and error made by the subject in order to succeed the task. However, a more detailed analysis has to be done in the future.

\textsuperscript{1}https://www.sqlite.org/
\textsuperscript{2}http://www.r-project.org/
Task #1

At first, we oppose our modalities for the completion time and the precision of the final placement. Unfortunately, the test shows that there is no significant difference between virtual reality and augmented reality for these two variables. Concerning the precision, we notice however a significant difference with a $p = .018$ when opposing dynamic and static (fig. 3.6). This result is confirmed by the test on the distortion of the ellipse, that returns a $p = .017$ for the opposition between dynamic and static (fig. 3.7).

Figure 3.6: Precision for task #1 according to the ability to move

Figure 3.7: Ellipse distortion in task #1 according to the ability to move

This suggests that the ability to move helps the subjects to perform the task. From the questionnaire, we know that ten of the twelve subjects who had tested both static and dynamic modalities preferred the dynamic modality, and that the two others were undecided. It emerges from the interview that they prefer dynamic essentially for practical reasons, mainly to have a better view of the whole scene because they can choose various points of view.

In contrast, the augmented reality does not seem to have a real influence. However, nine of the twelve subjects who had tested both virtual reality and augmented reality modalities preferred the augmented reality modality, and one was undecided. But this time, this is more for esthetic reasons. Only a few of them said that it is easier in augmented reality to get their bearings in the room.

In the questionnaire, we ask them their experience with interactive devices such as smartphones or tablet computers, and 3D object manipulation or augmented reality software. We make a median split both on their experience with devices and applications, and try to oppose the subjects with high and low experience. The statistical test does not show any significant results in terms of completion time or precision. It means that no special skills are required to use our application and even people that are not used with interactive devices or 3D or AR environment are able to perform the task without more difficulties. This result is confirmed by the questionnaire, because the subjects gave a grade of 3.68/4 for the ease of learn and intuitiveness of the application, and 3.59/4 for its ease of use.

We also tried to oppose static and dynamic for the precision of the final placement in the groups formed by the median split. The results are still significantly different for high experienced subjects, both with devices experience ($p = .026$) and applications experience ($p = .04$). However, we were surprised to notice that they are not for low experienced subjects. We yet don't know how to interpret this finding.
Task #2

Unfortunately, the data we got for the task #2 are not usable. Indeed, all the subjects have easily succeed the task in a very short time, regardless of the modalities. The reason is that the task was finally not difficult at all, due to time and technical constraints.

Originally, this task was supposed to show if the subjects are able to place virtual objects next to real objects (augmented reality modality) or if this is easier for them to have only virtual objects (virtual reality modality). So, this task should have been more focused on the difference between the two room representations.

At the beginning, we wanted to place some real pieces of furniture in the room, and the subjects should have to place the virtual pieces of furniture at each side of the real one. However, we had to discard this idea.

The first problem happens in augmented reality when a subject places a virtual object behind a real one. In this case, the virtual object should disappear because it is hidden by the real one. That is something difficult to do and we did not have enough time to develop such a feature.

Thus, we decided to place virtual boxes at the same positions as the real objects. The experience is already a bit different, but as the real objects remain in the room, it is still interesting.

But at the end we noticed the issue with the gyroscope (explained in section 2.2.3), so we had to remove the real objects from the room because the boxes are drifting away...

Finally, the task was only to place virtual objects next to other virtual objects in both modalities. Moreover, the collisions were provided with the walls and the existing objects, so that task was far too easy. Subjects confirmed this during the interview.
3.2.2 Qualitative analysis

The qualitative analysis is based on visualization of the log data along with the videotaped interaction.

Static

By observing the placement of the furniture and the ellipse that fit best the positions of the chairs in task #1, we can deduce the main problems that the subjects have encountered in static modality.

Often, as we see on figures 3.8a, 3.8b and 3.8c, the subject has some difficulties to get the depth of the room, so the ellipse is stretched in the direction of his gaze. Moreover, they tend to place the first piece of furniture very far, so everything (3.8c) or just the table (3.8b) is off-center. Another consequence we can notice in figures 3.8a and 3.8c, is that the three farther chairs are aligned. In figure 3.8a, they are aligned on the tape on the floor. Several subjects said that the tape was very helpful for them to place the furniture.

In figure 3.8d, we can see another problem. The subject has started by placing the table, two chairs on its side, and then two chairs in front of him. Thus, he was in trouble to place the remaining chairs because his view was obstructed.

![Figure 3.8: Final placement in static modality for task #1.](image)

The gray area is the area delimited by the tape. The pink star is the position of the stand. The square is the position of the center of the table and the circles those of the center of the chairs. Their color indicate the order of creation, from brown to white (from dark to light)
Dynamic

In the dynamic modality, also for task #1, it is interesting to look at the various displacement strategies the subjects have adopted.

In picture 3.9a, we can see that the subject stay at the middle of his circle of chairs, simply rotating on himself without moving. At the end, he moves out of his circle to have an external point of view and place the table in the middle. Finally, he moves a little to have another observation point.

The subject in picture 3.9b uses also only three points, but interacts in all of them. He immediately goes on a corner, place the table and the chairs farther from him. He places half of them, until they are to close. Then, he moves to the opposite corner of the room to place the remaining chairs. Finally, he comes back to his first position to make some adjustments.

The figure 3.9c is very different. There is much more displacement in one half of the area, but it is difficult to see a real strategy.

On contrary, the strategy on the figure 3.9d is very clear. The subject is turning around his table and stop regularly in order to place the two chairs opposite to him. Then, he did another round to make some corrections.

The strategy is the same in the figure 3.9e, but the circle of chairs is far bigger, so the subject uses all the available space.

To summarize, we notice that the strategy of displacement varies widely among the subjects. Some uses only a few positions to place the pieces of furniture, whereas others uses the whole space in a very dynamic way. And while many of the subjects stay inside their circle of chairs, some needs to have at least one external point of view.

We also noticed that nearly none of the subjects interacts while moving. They usually stay still to interact, then move to another position, and stop to interact from this other point of view. Most of the time, they keep away from the object with which they interact, so they have a larger view. This fact is confirmed in the questionnaire where sixteen of the twenty-four subjects said that the field of view (i.e. the camera angle) is not sufficient.

Finally, despite these far different strategies, there is not a single best one. Indeed, we did not find any remarkable relation between the strategy adopted and the completion time or the precision of the placement.
3.2. RESULTS AND ANALYSIS

CHAPTER 3. USER STUDY

Figure 3.9: Final placement in dynamic modality for task #1. The gray area is the area delimited by the tape in which the subject can move. The square is the position of the center of the table and the circles those of the center of the chairs. Their color indicates the order of creation, from brown to white (from dark to light). The line indicates the displacement of the subject (one point every 0.1 second), from red to yellow (from dark to light).
Chapter 4

Discussion

4.1 Limitations

I discuss the limitations we had, both in the application and the user study.

iPad application

One drawback of our setup is that we have to use the Kinect. I manage to do a marker free application, but I failed to compute the position of the iPad only with its sensors. The use of an external device imposes many constraints for the user, especially in his movements, and makes the room unnatural. To improve this point, we should try to do sensor fusion with all the sensors of the iPad.

Another limitation of my application is the performance. As I don't use any shaders or optimization in the OSG code, due to time constraints, it was difficult to keep a good frame rate. At first, I used virtual pieces of furniture made of simple models of Roombots, but as soon as two or three of these objects are present in the field of view, the frame rate goes under 10 fps, that is not acceptable. Thus, we decided for this first prototype to use models of furniture made of cubes.

For the same reasons, there is no shadow at all. That is a problem, because the virtual furniture seem to float, what is said to negatively influence the user experience.

I had also at the beginning a problem with the static libraries of OSG. By default, Xcode had compiled them in debug mode. I noticed later that compiling them in release mode improve the frame rate by a factor of 7.

The collision detection system used is not accurate, because the pieces of furniture are seen as spheres. This is sufficient for this first prototype and the user study, as no collision are provided between the virtual pieces of furniture added by the user.

The creation system is not perfect. The first idea was to use a “drag and drop” mechanism. The user press on the button corresponding to the desired piece of furniture and drag it on the main window where the virtual object appears stick to the user’s finger. Then, to avoid collision, a color feedback is given to the user. If the object is green, he can release is finger and the object is created. In case of collision, the object turns red and the user has to find another placement.

To develop such a feature, we were also limited by an issue in OSG about the color materials. As soon as I set the lights on, the color material is no more taken into account. So I could use color in the HUDs or with the arrow above the selected object because the are not sensitive to
4.2. FUTURE PERSPECTIVES

the lights, but not with the 3D objects of the scene... In this version of the application, the color on the pieces of furniture is given in the models themselves, and not in OSG.

There is also the problem mentioned at section 3.2.1 about the task #2. As it is difficult to hide a virtual object behind a real one, the room needs to be empty.

Finally, there is a lack of accuracy in our computation of the position and the orientation of the device. As I already explained in section 2.2.2 and 2.2.3, the position given by the Kinect is not accurate to the centimeter, and we use only an estimation of the position of the iPad from the position of the user's neck. For the orientation, the drift of the accelerometer is a huge problem. If the imprecision of the Kinect is acceptable, the drift of the gyroscope need to be corrected with a high priority. An idea for that is to use the other sensors of the iPad (at least the magnetometer) to confirm the value of the yaw rotation given by the gyroscope. This is feasible, as the magnetometer measures the yaw rotation according to our setup. But using the magnetometer induce some delays, so some tests are needed to find the best way to combine the two information.

User study

One of the limitations of our user study concerns the recruitment of the subjects. We had a low number of subjects, and they are all related to EPFL. And it was difficult to recruit students due to the exam period.

The data analysis turned out to be more complex than expected. Dealing with four conditions, two tasks and a questionnaire gave us a huge amount of data to analyze. Unfortunately, a lot are unusable, especially those from the task #2 that could have shown a significant difference between the two room representations, i.e. virtual and augmented reality.

The iPad application has still rather limited functionality, so that did not allow us to define more complex tasks. Moreover, the tasks we proposed were maybe too much open. If we had imposed some more constraint, for example a maximum completion time, we should have captured more significant differences between our modalities.

4.2 Future perspectives

Regarding the user study, some statistical work can still be done with the data collected. I already mentioned the number of action that has to be analyze more carefully, but it is the same for other data, such as some parts of the questionnaire or the voice recorded interview.

In general, it might be interesting to look more in details the sequence of the tasks, and not only the final placement. We could imagine to have a 2D animation in which we can see the user and the furniture moving in real time. Thus, we could have a better idea of the strategy adopted be the subjects and maybe see some other problems they had with the application, the task or the hardware.

For the iPad application, the most important will be to correct the drift of the gyroscope and find a way to compute the position of the iPad without the need of an external device.

Then it would be useful to improve the rendering of the scene. We should provide models of furniture made of Roombots and not of cubes. Moreover, it would be necessary to draw a simple shadow below the virtual objects in order to improve widely the realism of the scene.
We can also imagine new features, such as the creation of the pieces of furniture with a “drag and drop” mechanism that I already mentioned, or new interactions with the furniture. We could provide the ability to change the shape or the size of the furniture, either by scaling the passive parts of the structure, or by reconfiguring the whole Roombots structure.

It would be interesting to think about a way to provide a 2D view from above, as in classic interior design software. The users could place the furniture in the 2D view, and then see these virtual 3D objects with augmented reality. That is easy to develop it in a known room, but challenging in we need to reconstruct in real time the 2D view of an unknown room thanks to the camera view.

To go further, we could also imagine a 3D reconstruction of the room, using image processing algorithms as corner detection. Thus, the room will not need to be empty any more, because the real objects would be detected by the application and integrated in the virtual scene.

Finally, the most interesting perspective is to provide the link with the real robots. That is to arrange the room with virtual furniture thanks to this application, and then look at the real Roombots construct the real pieces of furniture. But this will require much more work, both on the application and on the real Roombots.
Conclusion

The iPad application developed in the scope of this project allows the user to arrange virtual 3D pieces of furniture either in a virtual reality environment or an augmented reality environment. To provide augmented reality on such a device, we need to know its orientation, which is given by the gyroscope the iPad, and its position, which is computed from the data of an external device, a Kinect.

The user study we conducted has shown that the ability to move inside the room (dynamic modality) helps the user to interact with the virtual furniture in terms of precision, but the use of augmented reality does not have a significant influence. During the interview, users said that they prefer the dynamic modality for practical reasons, and the augmented reality for esthetic reasons.

It can be concluded that a mobile interface with an augmented reality environment is more efficient to control modular robots than a traditional interface. Moreover, this kind of application is appreciated by the users.
Bibliography


Appendices
Appel à participation

Au laboratoire de Biorobotique (BioRob, EPFL) une nouvelle application utilisant la réalité augmentée a été développée dans le cadre du projet Roombots et nous souhaiterions l'évaluer par le biais d'une étude utilisateur.

Il s'agira d'une étude centrée sur les aspects d'interaction et de facilité d'utilisation, durant laquelle les participants devront utiliser le système et par la suite remplir un questionnaire. La durée globale de l'étude ne devrait pas dépasser 60 minutes.

Nous recherchons des étudiants toutes filières confondues. Aucune compétence particulière n'est requise.


Votre participation, récompensée par des bons pour un menu Roulottes (pizza ou kebab) ou équivalent, nous aidera à améliorer notre système.

Si vous êtes intéressés, merci de répondre à cet email et nous vous recontacterons prochainement.

Sincères salutations,

******************************************************************************

Call for participants

At the Biorobotics Laboratory (BioRob, EPFL) we are evaluating a new software application based on augmented reality in the scope of the Roombots project, and thus looking for participants to take part in a user study.

It will be a study focusing on aspects of usability and interaction where participants use the system and fill out a short questionnaire afterward. Overall, this will take about 60 minutes.

We are looking for students of all backgrounds. There are no special skills required.

The study will take place at BioRob, EPFL between December the 5th and December the 9th.

With participating, you will help us to improve our system and you will receive coupons for Roulottes menu (pizza or kebab) or an equivalent reward.

If you are interested, please reply to this email and we will get in touch with you soon.

Thanks a lot,

Jérémy Blatter (jeremy.blatter@epfl.ch)
Stéphane Bonardi (stephane.bonardi@epfl.ch)

---------------------------------------------
Stéphane Bonardi
Research Assistant - PhD student
EPFL - STI - IBI - BIOROB
INN 216 (bâtiment INN)
Station 14
CH-1015 Lausanne
Phone: +41 21 693 47 45
e-Mail: stephane.bonardi@epfl.ch
Web: http://biorob.epfl.ch/
---------------------------------------------
Informed Consent Form

BIOROB, EPFL is asking you to participate in evaluating a new application. By participating in this evaluation, you will help us improve this and other EPFL systems.

Please read this consent agreement carefully before agreeing to participate in the experiment.

Purpose of the experiment:
To study how people use the iPad to manipulate objects in a 3D environment.

What you will do in this experiment:
You are asked to solve a manipulation task in a 3D environment using the iPad. We will observe you and record information about how you work with the application. By signing this form, you give your permission to EPFL to use your voice, verbal statements, and videotaped pictures for the purposes of evaluating the application and showing results of these evaluations. We will not use your full name.

Time required:
The experiment will take approximately 60 minutes to complete.

Risks:
There are no anticipated risks associated with participating in this study.

Benefits:
You will receive 2 Roulottes coupons or equivalent for participating.

Confidentiality:
Your participation in this study will remain confidential, and your identity will not be stored with your data.

Participation and withdrawal:
Your participation in this study is completely voluntary, and you may withdraw from the experiment at any time without penalty by informing the evaluator.

Contact:
If you have any questions about this study, please contact Jérémy Blatter or Stéphane Bonardi, BIOROB, EPFL. Email: jeremy.blatter@epfl.ch, stephane.bonardi@epfl.ch

Agreement:
The purpose and nature of this research have been sufficiently explained and I agree to participate in this study. I understand that I am free to withdraw at any time without incurring any penalty.

Signature: ___________________________________________ Date: _______________________

Name (print): _____________________________________________________________________
I. Your Experience

1. How much experience have you had with the following types of interactive devices?

<table>
<thead>
<tr>
<th>Device Type</th>
<th>I never used it</th>
<th>I tried it out</th>
<th>I use it sporadically</th>
<th>I use it daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone / other smart phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>iPad / other tablet computer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. How much experience have you had with 3D object manipulation / augmented reality?

<table>
<thead>
<tr>
<th>Software Type</th>
<th>I never used it</th>
<th>I tried it out</th>
<th>I use it sporadically</th>
<th>I use it daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D modeling software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smart phone game</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Augmented reality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

II. What you think about the Application

3. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>Statement</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>I think, the <strong>hardware (iPad / stand)</strong> is easy to handle (on the stand)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is useful for the given task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is comfortable to use</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to rotate the stand.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I think, the software (application) ...</td>
<td>Totally disagree</td>
<td>Fully agree</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>is easy to use</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is useful for the given task</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is responding too slow</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is easy to learn</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>produces many errors</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>The camera (angle) is big enough.</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>It is helpful to have a virtual</td>
<td></td>
<td></td>
</tr>
<tr>
<td>representation of the room.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to see the</td>
<td></td>
<td></td>
</tr>
<tr>
<td>room in real.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

III. Your Preferences

4. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>I think, the software (application) ...</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>is frustrating</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is confusing</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is intuitive</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>I find it easy to get the system to</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td>do what I want it to do.</td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>It would be easy for me to become</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td>skillful at using the system.</td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>The tasks ...</td>
<td>Totally disagree</td>
<td>Fully agree</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------</td>
<td>------------</td>
</tr>
<tr>
<td>were difficult</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>were not clear to me</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>were fun</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

5. Overall experience:

Which condition did you prefer?  ☐ Virtual room  ☐ Real room

Why?

What was NOT good? 😞  What was GOOD? 😊

Thank you!!
## I. Your Experience

1. How much experience have you had with the following types of interactive devices?

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Never Used</th>
<th>Tried Out</th>
<th>Use Sporadically</th>
<th>Use Daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone / other smart phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>iPad / other tablet computer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. How much experience have you had with 3D object manipulation / augmented reality?

<table>
<thead>
<tr>
<th>Software Type</th>
<th>Never Used</th>
<th>Tried Out</th>
<th>Use Sporadically</th>
<th>Use Daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D modeling software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smart phone game</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Augmented reality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## II. What you think about the Application

3. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>Statement Description</th>
<th>Totally Disagree</th>
<th>Fully Agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is easy to handle (holding it)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is easy to handle (on the stand)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is useful for the given task (holding it)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is useful for the given task (on the stand)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is comfortable to use (holding it)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is comfortable to use (on the stand)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to move around the room</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to rotate the stand</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Remarks:**
I think, the **software (application)** ...

<table>
<thead>
<tr>
<th></th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>is easy to use</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is useful for the given task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is responding too slow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is easy to learn</td>
<td></td>
<td></td>
</tr>
<tr>
<td>produces many errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The camera (angle) is big enough.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to have a virtual representation of the room.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

III. Your Preferences

4. How far do you agree to the following statements?

I think, the **software (application)** ...

<table>
<thead>
<tr>
<th></th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>is frustrating</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is confusing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is intuitive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I find it easy to get the system to do what I want it to do.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It would be easy for me to become skillful at using the system.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The **tasks** ...

<table>
<thead>
<tr>
<th></th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>were difficult</td>
<td></td>
<td></td>
</tr>
<tr>
<td>were not clear to me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>were fun</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. Overall experience:

Which condition did you prefer?  
   - iPad on stand  
   - moving in room  

Why?

What was NOT good? 😞  
What was GOOD? 😊

Thank you!!
I. Your Experience

1. How much experience have you had with the following types of interactive devices?

<table>
<thead>
<tr>
<th>Device Type</th>
<th>I never used it</th>
<th>I tried it out</th>
<th>I use it sporadically</th>
<th>I use it daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone / other smart phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>iPad / other tablet computer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. How much experience have you had with 3D object manipulation / augmented reality?

<table>
<thead>
<tr>
<th>Software Type</th>
<th>I never used it</th>
<th>I tried it out</th>
<th>I use it sporadically</th>
<th>I use it daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D modeling software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smart phone game</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Augmented reality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

II. What you think about the Application

3. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>Statement Description</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is easy to handle (holding it)</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is easy to handle (on the stand)</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is useful for the given task (holding it)</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is useful for the given task (on the stand)</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is comfortable to use (holding it)</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>I think, the <strong>hardware</strong> (iPad / stand) is comfortable to use (on the stand)</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>It is helpful to be able to move around the room.</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>It is helpful to be able to rotate the stand.</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>I think, the <strong>software (application)</strong> ...</td>
<td>Totally disagree</td>
<td>Fully agree</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>------------------</td>
<td>------------</td>
</tr>
<tr>
<td>is easy to use</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is useful for the given task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is responding too slow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is easy to learn</td>
<td></td>
<td></td>
</tr>
<tr>
<td>produces many errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The camera (angle) is big enough.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to see the room in real.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

III. Your Preferences

4. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>I think, the <strong>software (application)</strong> ...</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>is frustrating</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is confusing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is intuitive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I find it easy to get the system to do what I want it to do.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It would be easy for me to become skillful at using the system.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The <strong>tasks</strong> ...</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>were difficult</td>
<td></td>
<td></td>
</tr>
<tr>
<td>were not clear to me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>were fun</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. Overall experience:

Which condition did you prefer?  
- [ ] iPad on stand  
- [ ] moving in room  

Why?

What was NOT good? 😞  
What was GOOD? 😊

Thank you!!
I. Your Experience

1. How much experience have you had with the following types of interactive devices?

<table>
<thead>
<tr>
<th>Device</th>
<th>I never used it</th>
<th>I tried it out</th>
<th>I use it sporadically</th>
<th>I use it daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone / other smart phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>iPad / other tablet computer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. How much experience have you had with 3D object manipulation / augmented reality?

<table>
<thead>
<tr>
<th>Application</th>
<th>I never used it</th>
<th>I tried it out</th>
<th>I use it sporadically</th>
<th>I use it daily</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D modeling software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smart phone game</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Augmented reality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

II. What you think about the Application

3. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>Statement</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>I think, the hardware (iPad) is easy to handle (holding it) remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is useful for the given task remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is comfortable to use remarks:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to move around the room. remarks:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## III. Your Preferences

### 4. How far do you agree to the following statements?

<table>
<thead>
<tr>
<th>I think, the <strong>software (application)</strong> ...</th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>is easy to use</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is useful for the given task</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is responding too slow</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is easy to learn</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>produces many errors</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>The camera (angle) is big enough.</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>It is helpful to have a virtual representation of the room.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>It is helpful to be able to see the room in real.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is frustrating</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is confusing</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>is intuitive</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>I find it easy to get the system to do what I want it to do.</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
<tr>
<td>It would be easy for me to become skillful at using the system.</td>
<td>Virtual room</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real room</td>
<td></td>
</tr>
</tbody>
</table>
The tasks...

<table>
<thead>
<tr>
<th></th>
<th>Totally disagree</th>
<th>Fully agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>were difficult</td>
<td>☐</td>
<td>☑</td>
</tr>
<tr>
<td>were not clear to me</td>
<td>☐</td>
<td>☑</td>
</tr>
<tr>
<td>were fun</td>
<td>☑</td>
<td>☑</td>
</tr>
</tbody>
</table>

5. Overall experience:

Which condition did you prefer? ☐ Virtual room  ☐ Real room

Why?

What was NOT good? 😞  What was GOOD? 😊

Thank you!!
User action log file example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple/DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
  <array>
    <string>2.944618 CRE 0 1 -2.960046 0.011729 0.000000 0.007820</string>
    <string>2.944626 SEL 0</string>
    <string>5.317025 MOV 0 -2.606069 0.016728 0.000000</string>
    <string>4.383286 ROT 0 0.150000</string>
    <string>5.487634 CCO 0 3</string>
    <string>6.384367 DSL 0</string>
    <string>8.384475 CRE 1 2 -1.364754 0.273645 0.000000 0.007820</string>
    <string>8.384543 SEL 1</string>
    <string>8.974623 MOV 1 -1.661759 -0.001746 0.000000</string>
    <string>9.246385 ROT 1 6.263185</string>
    <string>10.857463 DSL 1</string>
    <string>14.428076 TYP 0 13</string>
    <string>14.428135 MOV 0 -2.606069 0.016728 0.000000</string>
    <string>14.428173 ROT 0 0.150000</string>
    <string>14.428218 TYP 1 21</string>
    <string>14.428250 MOV 1 -1.661759 -0.001746 0.000000</string>
    <string>14.428284 ROT 1 6.263185</string>
  </array>
</plist>
```