UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

WHITE HAT SNIPER
Future User Interface

Ertan Dağıstanlı, Roberto Sanchez, Leutrim Kaleci

ertan.dagistanli@unifr.ch,robertogonzalo.sanchezalban@unifr.ch,
leutrim.kaleci@unifr.ch
Future User Interfaces (FUI): multimodal, plastic and natural
Department of Computer Science, University of Fribourg

ABSTRACT

This paper describes the use of the multimodal interaction in our game called “White Hat Sniper”. The player controls a sniper rifle scope with the possibility of changing his field of vision. The design of the game permits players interact in two ways, either by using gesture and voice, or the conventional mouse and keyboard. At the end, we present the user evaluation in order to see the performance of each of the combinations, and to come up with some conclusion about using the multimodal interactions in our game.

1. INTRODUCTION

Nowadays, the designers have put the focus on designing interactions through which the users would interact with the computer in a more natural way. Basically, the main idea is using the multimodal interactions, which based on [1] means: “interaction with the virtual and physical environment through natural modes of communication”. As is stated in [2], “in everyday life, during natural human-human communication, human beings use the five senses of touch, hearing, sight, smell, and taste in order to interact with external world”. So, the main idea of the multimodal interactions is to use, as much as possible, the interactions between human-human communication into the human-computer communications. According to this idea of multimodal interaction, we have designed a game, and we have applied there some of multimodal interactions, such that the players would have the possibility to

play the game in a more natural way. So, in this paper, firstly, we give you a brief description to the game, in section 2, afterwards, in section 3, we describe the two possible ways to play the game. Respectively, here we describe the two combinations of modalities used for playing the game. In order to have a better understanding about that that how these modalities are handled by the system, and by the user's experience as well, while a player playing the game, in section 4, we do this analysis based on CASE model and CARE model. In order to make a comparisons between the two combinations of modalities, we have done an evaluation, which in more details is described in section 5. At the end, in section 6, we describe some of our issues we faced during the evaluation, and our future thoughts regarding the multimodal interactions.

2. THE GAME

The design of game consider the possibility to combine two actions in a synergistic fashion, also the possibility of concurrence according with the CASE model. During this project, we attempted to recreate a game where the player feels free to explore into a virtual world. Some gameplay elements were considered:

- **Stage**: The player is located in an upper part of a building and some targets appear in a random fashion through a big space, it enables the player make actions like: zooming, pointing and shooting.
- **Freedom of movement:** The user is able to move the scope of the riffle in 3D dimensions, and also in our first design, the character could walk into the stage with the voice commands "go" and "stop". This latter implementation was not implemented because it implied more details around the stage therefore we decided to keep simple, and only the movement of the scope was implemented.
- **Controls**: The usual way to play this kind of game is the keyboard and mouse, we want to find a new way to play this kind of games. We include in our design gesture and voice, such a way that the pointing and zooming action are done by gesture with the hand, and the command of fire by voice commands.

The principal actions in the game are: Zooming, shooting and pointing, we use the different modalities to manage these actions.

2.1. The Description of the Game

In this sniper game, the player uses a scope for pointing the objectives, the target objects are in yellow color while the obstacles are in red color. The objective of the game is to destroy all the yellow target without destroying the obstacles. We measure the time that the user spend for each shooting action and the number of bullets that the user takes in order to destroy five objectives. These measurements permit the comparison between two modalities in term of performance, efficiency and effectiveness.

2.2. The Architecture of the Game

For the purpose of the design and the implementation of the game we use Unity 3D platform, this platform includes some libraries in order to implement the different modalities. We recognize the gesture of the user through XBOX ONE (Kinect sensor) and voice by Windows voice recognizer.
The kinect V2 SDK provides an simple abstraction of the human body, where each node is a part of the body and Windows voice recognizer permit to implement a desirable grammar according with the game.[3,4] .

The game was developed using Gameobjects in Unity 3D where we adapt the behavior of each object using scripts. We include the library of V2 SDK in Unity 3D platform and we get the body skeleton of the player, using some triangulation and measurements of the body, at every frame we calculate the position and rotation of the scope's riffle. Our design consider the measure the distances between head, hands and hips. We do not consider the use of shoulders' position because sometimes they are hidden by the hand movements of the player (we realized that fact during the implementation). We implement the principal actions of the game in different scripts because it permit to run each one in parallel, this fact was important in the moment to implement the synergistic action "pointing and zooming".

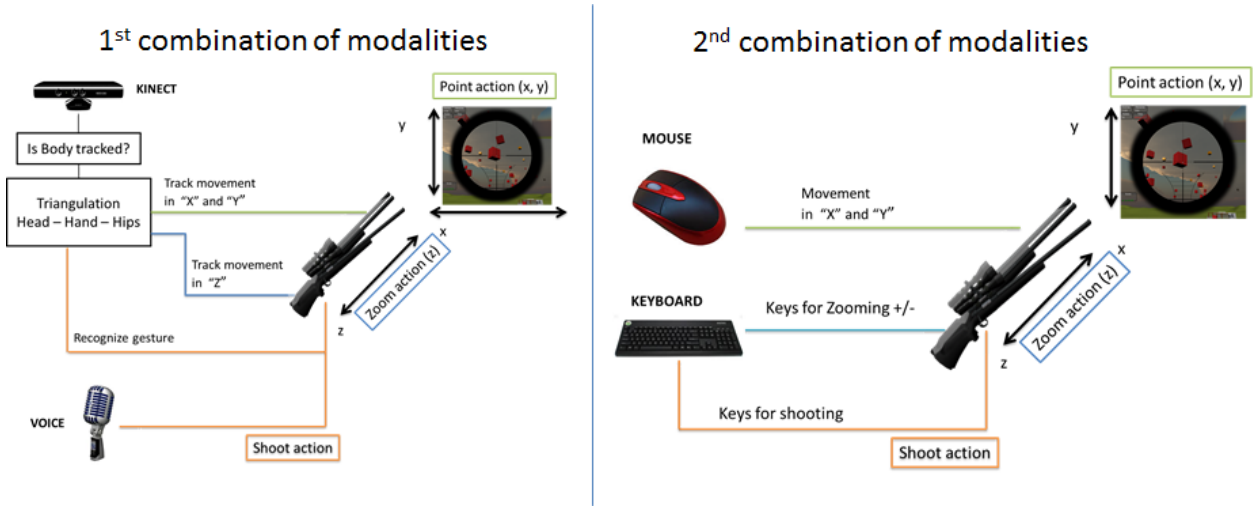


Figure 1. Architecture of the game

3. MULTIMODAL INTERACTIONS

In this section we are going to describe the two combinations of the modalities. Respectively, we are going to describe the possible interactions while playing the game.

3.1. The 1st Combination of Modalities

In Figure 1 we describe the interaction with the game. Once the body is detected, we get skeleton of the body of the player, we use the tracking of the right hand, and we make a triangulation between hand, head and the position of the kinect for knowing the relative coordinates (x, y, z) of the right hand. Using some comparisons and some deadbands with respect to axes X and Y, we

perform the action of pointing. Using the movements with respect to the axe Z, we change the field of view of the rifle's scope. Thus with gesture we perform pointing and zooming actions

We implement the voice recognition to recognize specific words like: "zoom in", "zoom out", "go", "stop", "fire" to perform their correspondent actions in the game. In our first design the words "go" and "stop" performed to move the sniper character though the scene, it makes the player be involved in a virtual world however in the last design we decided do not use these commands because it implied more 3D development in the application.

3.2. The 2nd Combination of Modalities

We implement similar actions than before by using the keyboard, the mouse and the voice, we can appreciate the principal actions in Figure 1. Keyboard is used to perform the following actions: shooting and zooming action in the rifle, we use the words: "Fire", "zoom in", "zoom out" respectively. Mouse is used to perform the action of pointing the rifle scope, if the user decide to use voice, the user can perform the similar actions described above.

4. CASE MODEL & CARE MODEL

One of the most important thing in multimodal interfaces is to know how the interactions are handled by the architecture of the system itself, in our case the game, and by the users' point of view while interacting with the game system. Basically, this analysis is possible by analyzing the properties of the CASE model, which consider the machine side, and the properties of the CARE model, which considers the user's side (usability properties).

4.1. CASE Model

Interactions performed with regard to the properties of the CASE model are analyzed in following:

- **Concurrent:**

- An example of the actions running in parallel, i.e. being performed concurrently, is when the player can change the position of the target by using a gesture, resp. the mouse, and at the same time shooting to the objects with the voice command, resp. keyboard.

- Another example of this property is when the player shoots whether using the gesture command, or voice command, the player can perform zooming and pointing at the same time.

- **Synergistic:**

- We have two action for locating an objective in the application as we show in Figure 2, the first action is pointing and the other action is zooming. We consider that both

actions can be managed in a synergistic fashion. In the case of keyboard and mouse we combine the movement of the mouse in the axes X and Y, and when the user want to do a zooming in the application, the user uses the keyboard (key1 for zooming in, and key2 for zooming out). In the case of gesture we can manage both actions at the same time, we use the movement in X and Y for pointing and the movement in Z for zooming, this combination is more intuitive and natural for the user, and depending of the movement one of them is dominant.

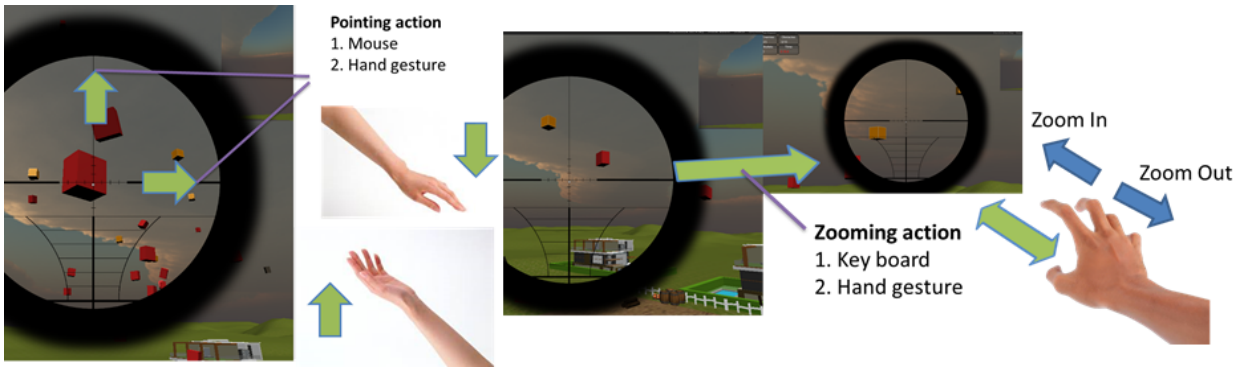


Figure 2. Pointing and zooming action

- **Exclusive:**
- In our game, we consider the command "restart" as an exclusive action because during this action, not pointing or zooming is permitted. Once the action is accomplished then we can continue with the normal operation.

4.2. CARE Model

Regarding it that how our game deals with the properties of the CARE Model is described in the following by showing some of the examples considering to each property:

- **Complementary:**
- *Mouse AND Voice*: In our game, an example of complementary situation might be when a player wants to destroy an object, so the player moves the mouse to find the desired target, and afterwards, the player shoots towards the targeted object using the voice command "fire". So, here we need the use of two modalities, in order to perform the action of destroying an object.
- **Assignment:**
- *Mouse*: Example of this property in our game is only used in the 1st combination of modalities, so there you can use only the mouse to change the position of the target.
- **Redundancy:**
- *Keyboard AND/OR Voice*: you can shoot either using voice or mouse, or both of them
- *Keyboard AND/OR Voice*: you can zoom in or zoom out by using either the keyboard or the voice, or both of them.

- *Gesture AND/OR Voice*: you can shoot either by using the gesture for shooting or by using the voice command "fire"
- *Gesture AND/OR Voice*: you can zoom in or zoom out by using either the keyboard or the voice, or both of them.

- **Equivalence:**
- *Keyboard OR Voice*: you can shoot either using voice or mouse
- *Gesture OR Voice*: you can shoot either using gesture or voice
- *Gesture OR Voice:* you can zoom in or zoom out either by using the gesture or voice

5. EVALUATION

In order to compare the two different modalities mentioned in section 3, we did an evaluation. So, in this section, we are going to describe this evaluation. Firstly, we are going to give you a description about the evaluation, and afterwards describing the results we ended up from the evaluation.

5.1. Evaluation Description

We evaluated the system using four users. The evaluation was done within group, i.e. each user was asked to do the same task with both of the combinations of the modalities to play the game. So, the users were asked to destroy five yellow objects, once using the 1st combination of modalities, and then doing the same task using the 2nd combination of modalities.

Our *hypothesis* was that:

- 2nd combination of modalities is better for playing the game, i.e. achieving better results than the 1st combination of modalities.

The *independent variable* was:

- 1st combination of modalities vs 2nd combination of modalities, it means destroy five yellow targets.

The *dependent variables* were:

- The time that the player needed to destroy a given number of objects
- The number of bullets that the player needed to destroy the objects
- The satisfaction using one of the combinations of the modalities, Rating: 0 - 5

At the end, the users have been asked to give some comments regarding the experience t hey had while using these two different type of interactions with the game.

5.2. Evaluation Results

The results taken out from the evaluation are shown in figures below. The results for the 1st interface (i.e. the 1st combination of modalities) are shown in Table 1, and the results of the 2nd interface(the 2nd combination of modalities) are shown in Table 2.

Table 1. The results of the 1st interface

Users	Bullets	Time	Satisfaction

User1	39	106 s	3
User2	10	78 s	2
User3	11	94 s	2
User4	16	145 s	3

.

Table 2. The results of the 2nd interface

Users	Bullets	Time	Satisfaction
User1	6	12 s	2
User2	5	14 s	4
User3	7	15 s	5
User4	5	11 s	5

Whereas the average performance regarding the 1st interface is shown on the graph in Figure 2, whereas the average performance regarding the 2nd interface is shown on the graph in Figure 3.

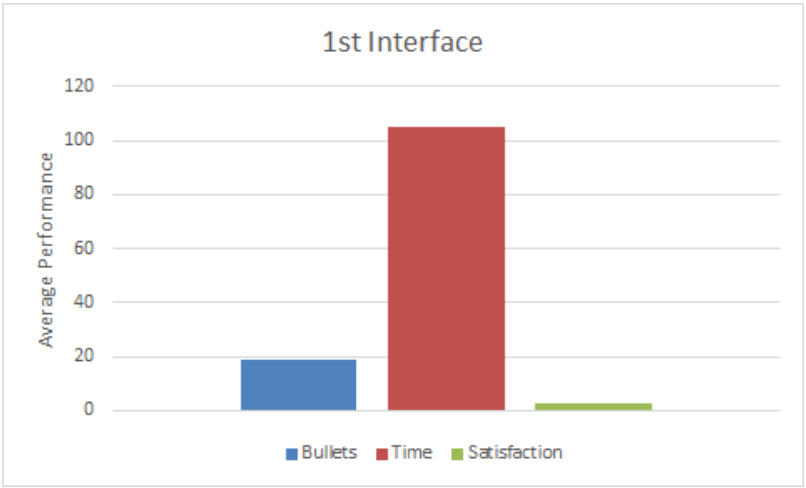


Figure 3. The average performance of the 1st interface

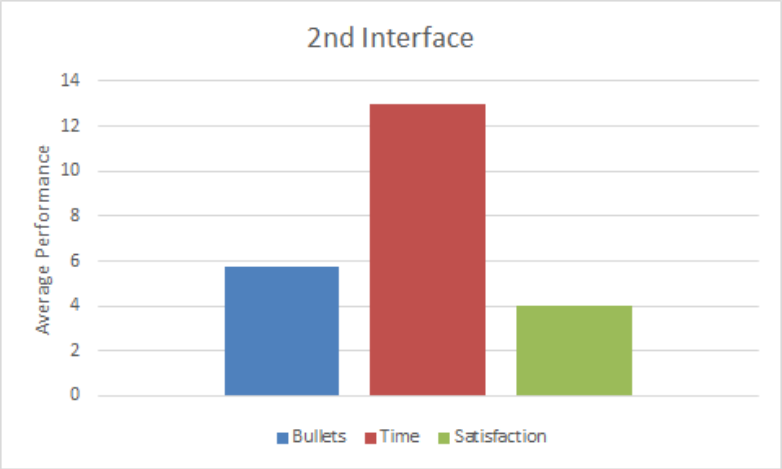
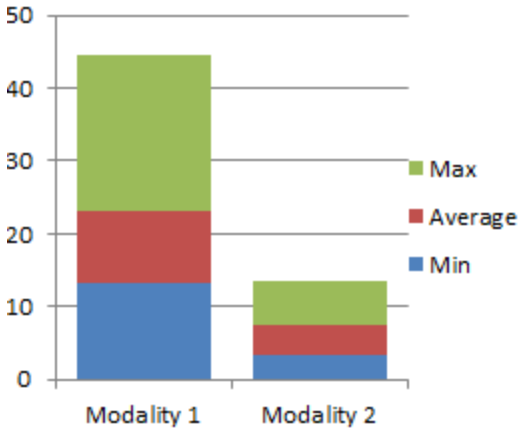


Figure 4. The average performance of the 2nd interface

So, the Figure 3 and the Figure 4 show the average performance for each dependent variable that we used to see the performance of the both interfaces, and such that being able to see the differences between each other. Also, it is obvious that the users have performed better while playing with the 2nd interface.

About the time that the user takes for destroying an object with gesture is from 3 up to 4 times slower than destroying by mouse and keyboard, as is shown in Figure 5.



	Average	Min	Max
Interface 1	23.1	13.3	31.2
Interface 2	7.5	3.36	10.1
Factor	3.08	3.96	3.09

Figure 5. Time performance, interface 1 vs. interface 2.

When the users were asked about giving some comments, generally they mentioned that they found it hard to adapt to the new interface (the 1st combination of modalities), and that it takes time to adapt. They found it interesting and challengeable playing with such a new interface, but their satisfaction of achieving great results as using the other interface were not satisfiable. For instance, they had problem with the voice interaction, since it is taking a while until this is processed, and such that the action is taking a bit time in order to happen. Also, they were complaining regarding the gesture interactions, because after a while, they say, they get tired.

Regarding the voice command, they were saying, for example, “fire” for shooting the bullets, but they were not sure that the command is going to be processed, so we realised that we needed kind of feedback for that, like having something that would appear on the screen that would tell to the player that the command is being processed (e.g. “You said: “fire”).

6. CONCLUSION

We conclude that using kinect V2 and the abstraction of body nodes, we can include more triangulations in order to recognize better the gestures, in this project we use basic comparisons in order to recognize the gestures, but the technology provide the measurement of the distance between each part of the body, therefore we can built an algorithm that recognize some patterns of movements.

Regarding to user’s experience, about voice recognition, we observed that user thinks that if they speak slowly, faster or loudly, that would be reflected also in the application. It could be also as an improvement include in the project the recognition of the voice intonation for some purposes like speed in shooting.

With respect to naturally, even though the keyboard is more precise, some users prefer the pointing and zooming action with the hands because is a little bit more natural and the user enjoys this interface, but the disadvantage is that the user gets tired quickly, and some emotions influence in the movements that they perform, so for instance they try to shoot with the same hand that they use for pointing, or some of them get confused between right and left hand.

The command “fire” by voice is slower than pressing a key by keyboard, therefore the user try to guess the future position of the target in order to be precise in the time and guarantee the destruction of the target. Thus a voice command for this purpose is not recommendable.

About CASE Model, we obtain a good synergistic action between pointing and zooming action using gesture, the user can do both activities at the same time, and depending on the movement one of the two actions is more predominant, but this synergistic action is not enough precise and quicker than the use of keyboard and mouse because is from 3 up to 4 times slower.

So, after doing the evaluation, we concluded that there still a lot of things that need to be improved. Basically, during all this experience of designing multimodal interactions, we experienced all the nowadays’ existing challenges in multimodal interfaces, such that we proved somehow to ourselves that they are really challengeable. For instance, regarding the voice, there are needed quite enough feedbacks, in order to assure the user that he/she has used the right command, but this still remains the challengeable. Also, regarding the gestures command, we found out that it is still challengeable to define a specific gesture. So, there is still place enough place for improvements in this area.

Finally, we conclude that if the consideration of accuracy and efficiency are important then the user prefers the combination: “mouse + keyboard+voice”.

7. REFERENCES

[1] Bourguet, M.L.(2003). “Designing and Prototyping Multimodal Commands”. Proceedings of Human-Computer Interaction (INTERACT'03), pp. 717-720.

[2] Meersman, Robert, et al. "On the Move to Meaningful Internet Systems: OTM 2014 Workshops." *On the Move to Meaningful Internet Systems: OTM 2014 Conferences*. Vol. 8842. Springer, 2014.

[3] Kinect library for Unity 3D:
http://forum.unity3d.com/threads/kinect-v2-with-ms-sdk.260106

[4] Configuration for kinect in Unity 3D:
http://channel9.msdn.com/coding4fun/kinect/Two-for-Kinect-4-Windows-v2-and-Unity-3D